BARRINGER

W.D. #5516D/0457A

CONTRACT REPORT NUMBER 59219-01-SQ

REQUISITION NUMBER 23440-5-9219

CONTRACTOR: BARRINGER RESEARCH LIMITED

TITLE: CANMET SULFUR-FORM ANALYZER
OPERATING MANUAL

CANMET SULFUR-FORM ANALYZER
OPERATING MANUAL

16 pp

DSS File No.: 07SQ.23440-5-9219-01-SQ
BRL File No: 303-11

August 1988

I. HARDWARE

The hardware in the system consists essentially of a Leco SC-132 Sulfur Analyzer and a Compaq microcomputer to acquire data from the Leco and manipulate it.

The Leco analyzer itself contains a microcomputer and an anolog-to-digital converter to input voltage signals to the computer. Several different analog signals need to be converted so a 16-channel analog multiplexer is used ahead of the ADC. (See the schematic of the ADC card, page S133 in the Leco operating manual). The signal of interest is the voltage from the SO_2 detector which appears at the channel 3 input to the multiplexer (test point TP3). The Leco instrument was modified so as to make this voltage accessible to the Compaq computer. The modification consisted of attaching a wire to TP3 and a wire to TPO (ground) on the ADC card and bringing these wires out to a connector on the front panel of the electronics drawer of the Leco. The signal from this connector can go to the ADC in the Compaq computer.

The ADC in the Compaq computer is a Data Translation DT2811-PGH which accepts input ranges of 0 to 5V, -5V to +5V, or -2.5V to +2.5V. The output voltage of the SO_2 detector in the Leco is typically about 8.5V when no sample is being analyzed. (The Leco manual suggests it be adjusted if it goes outside the range 7.5 to 9.2V). During a sample run, as SO_2 passes through the detector, this voltage may decrease to a level of 5 or 6 volts. This discrepancy in the output voltage of the Leco and the input voltage of the DT2811-PGH ADC made it necessary to shift the voltage. A voltage level shifter was designed and interposed between the Leco and the ADC. The schematic for this level shifter is shown in Figure 1. It has the function of lowering the Leco voltage so it suits the range used on the DT2811 (0-5V). There is a trimpot in the level shifter circuit

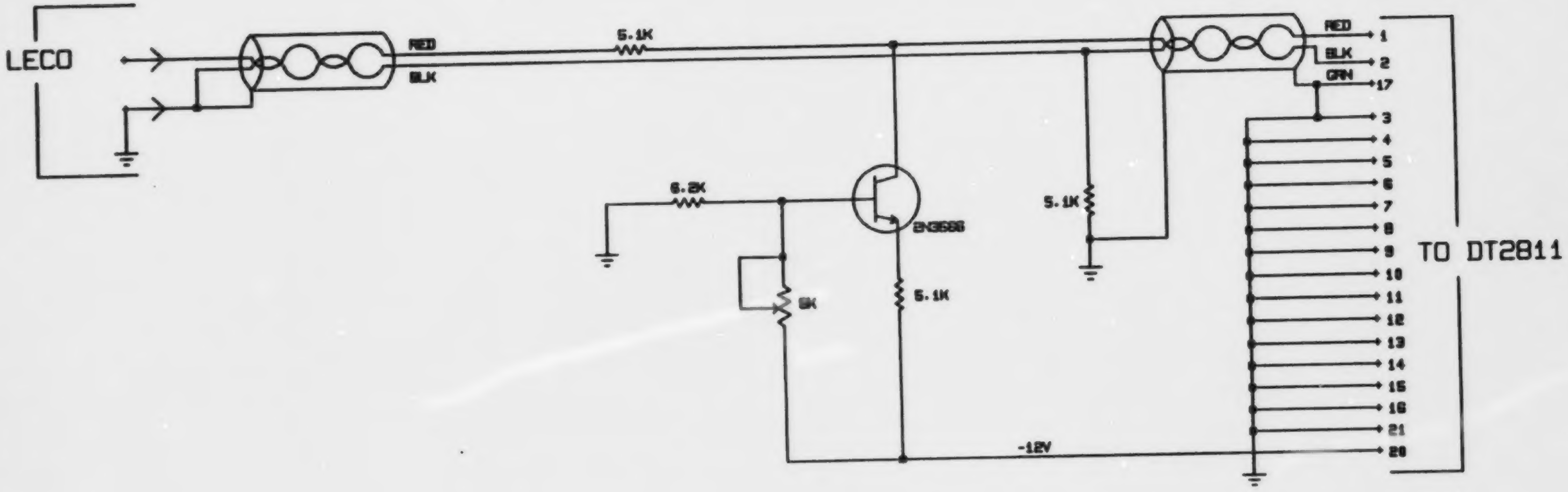

LECO
RED
BLK
5.1K
6.2K
2N3566
5K
5.1K
5.1K
-12V
RED
BLK
GRN
1
2
17
3
4
5
6
7
8
9
10
11
12
13
14
15
16
21
20
TO DT2811

FIGURE 1

Leco Output Voltage Level Shifter

which adjusts the amount of shift. This should not have to be adjusted unless the gain or zero for the IR cell in the Leco is changed (page S17 in the Leco Manual). Thus, there is a cable from the Leco to the level shifter circuit and another cable from the level shifter to the DT2811, which resides in the Compaq computer.

The Compaq computer is a Compaq Deskpro 286 which uses an Intel 80286 microprocessor and an Intel 80287 math coprocessor. It has 640k bytes of memory and runs DOS version 3.10. It has a 30 Megabyte hard disk drive, a 1.2 megabyte floppy disk drive, and a 360 kilobyte floppy disk drive. An Epson FX-86e printer is used to list programs and data as well as provide hard-copy output for graphics that appear on the monitor.

II. SOFTWARE

A. Introduction

The software on the Compaq computer operates under MS-DOS version 3.10 (Microsoft Disk Operating System). All information (programs and data) is stored in the form of files. These files are organized into separate directories. A hierarchy can exist so that a directory can have other directories under it. This flexibility allows the user to logically organize his files so that similar files are kept together. For details on how to interact with the computer through DOS, consult the MS-DOS Reference Guide.

Generally speaking, there are three kinds of programs used in this project: data acquisition programs, data manipulation programs, and utility programs. These classes of programs will be described in more detail in the following sections. Programs for this work have been written in three computer languages: C, FORTRAN and BASIC. The data acquisition programs acquire raw data and store them in data files. The data manipulation programs use the raw data files as input and may create new data files. All of these data files are kept in one directory of the hard disk (the DATA directory).

B. Data Acquisition Programs

The analog-to-digital converter card is a Data Translation model DT2811 PGH. This card was purchased with software that allowed data acquisition in a strip chart format. The original program from Data Translation is called LPSCHART. This was written in Microsoft C. The acquisition programs used in this project were derived from LPSCHART by making modifications to the original source code. The two programs that are normally used for data acquisition are ACQFIX and ACQFREE. ACQFIX acquires a fixed number of points (300) and then stops automatically.

ACQFREE acquires points until the operator stops the acquisition. There are two FORTRAN programs that are related to data acquisition. The first program, LINEAR, takes a raw data file as input and applies a linearization to the entire spectrochronogram to compensate for the non-linearity of the detector. It creates a new file as its output. The second program, ADJUST, applies the linearization and also applies the calibration factor.

Synthetic spectrochronograms were generated with the program FAKE, which was also a modification of LPSCHART. The program FAKE creates a data file that is in the same format as a normal data file, but the spectrochronogram is generated from a combination of 2 or 3 pure Lorentzian peaks plus a baseline. The parameters of the Lorentzians, i.e., mean, width, and amplitude, as well as the baseline are specified in the source program, FAKE.C at around line 740. In order to generate a different synthetic spectrochronogram, the parameters in FAKE.C are changed and FAKE is recompiled.

C. Data Manipulation Programs

The data manipulation programs are written in Microsoft FORTRAN. The software that deconvolutes the spectrochronogram using a model that contains two peaks is in the directory \FORT. There is similar software that uses a 3-peak model and that software is in directly \FORT\3PEAKS. There is also a version of the deconvolution program that uses a 3-peak model that asks the operator to estimate the positions of the three peaks rather than estimating them by itself. This program and its related files are in directory \FORT \3PEAKS \OPSET. There is also software using a 5-peak model, and that is in directory \FORT\5PEAKS.

The main deconvolution program in each subdirectory is called DECON. That is, the source file is called DECON.FOR and the file that is actually executed is called DECON.EXE. DECON uses the subroutines RDFILE, ESTIM, CURFIT, MATINV, FDERIV, FCHISQ, FNCTN, WRFILE and RDCAL.

All versions of the DECON program ask the operator for the name of the file that is to be used as input. It then fetches that file from the \DATA directory, and performs the deconvolution on it. The result of the deconvolution is a "fitted" spectrochronogram. This fitted spectrochronogram is written to the \DATA directory as a file. The fitted file has the name of the original file with an "F" appended to the beginning. For example, if the input file is A701.130, the fitted file is given the name FA701.130.

D. Utility Programs

A number of programs have been created that provide useful functions for anyone working with spectrochronograms. The programs ACQFIX and ACQFREE are the ones normally used to acquire data. They plot the spectrochronograms as it is being acquired and have the capability to display previously acquired chronograms. However, their display uses a low resolution graphics mode (320 by 200 points). In order to see the subtle details in the shape of chronograms, a program was written to plot spectrochronograms in high resolution mode (640 by 200 points). This program was written in BASIC and is called PLOT.

It is sometimes useful to see the actual numbers that were acquired from the ADC. The program DDATA (short for Dump Data) prints out to the screen (or the printer) the data in any specified file. This can be a raw data file, a fitted file, or even a file containing a synthetic spectrochronogram.

The AREA program takes any input file and prints out the accumulated total area at every data point in the spectrochronogram. The area is measured starting at the 41st point in the chronogram. The baseline is taken as the average level of the first 40 points.

E. The \DATA Directory

All spectrochronograms are acquired into files that are kept in a directory called \DATA. All the data acquisition programs ask the operator for a file name for the data to be stored before acquisition is begun. Much of the work in this project was done on Leco standards. The following convention was adopted for the naming of data files. The last three digits of the Leco standard number form the main part of the file name. Every filename can have a three character extension. The first spectrochronogram for a particular standard or sample would have an extension of .001., the second one, .002, etc. Thus, for Leco standard 501-701, the first acquired spectrochronogram is stored in file \DATA \701.001. A raw data file unchanged from the way it was originally acquired would be named like this. After a file has been acquired, a copy of it is made which is adjusted to compensate for the non-linearity of the detector and to apply the calibration. This adjusted file has the same name as the original file with an "A" appended to the beginning. Thus A531.015 is the adjusted version of 531.015. A deconvolution would result in a fitted file. The fitted file for A531.015 is called FA531.015. This is the result of the 15th run using Leco standard 501-531.

Synthetic spectrochronograms are generated by the program FAKE using equations only. These fake chronograms are stored in files having the exact same format as real chronograms. A different kind of fake chronogram can be generated by changing the parameters in the FAKE program. The files containing the fake chronograms are called FAKEX.001, where X is a number. Deconvolutions were also done on the fake files. The result of the deconvolution on FAKE40.001 would be in file FFAKE40.001.

The calibration of the instrument should be done daily. It is done by acquiring spectrochronograms of Leco 501-701 at 550°C, using no silica. These special calibration acquisitions are stored in files whose names begin with C701. The program LINEAR is used to linearize these spectrochronograms. The results of this are in files AC701.

The program CAL uses these calibration files AC701 as input to find a multiplication factor which is used to adjust all normal spectrochronograms. The current multiplication factor is always stored in the file \DATA\MULT.FAC.

The non-linearity of the detector is compensated for by means of a calibration curve which relates the detector response to a known amount of sulfur. This calibration curve was constructed using data acquired for benzenesulfonic acid. The benzenesulfonic acid spectrochronograms are in files named BSA.

III. ANALYTICAL SYSTEM OPERATION

A. MATERIALS

(1) Silica (Fisher Scientific Ltd.); fine grain 240 mesh floated silica powder

(2) 4-amino benzene sulfonic acid (Fisher Scientific Ltd.); ACS grade (99.9% pure) amino benzene sulfonic acid (ABSA; sulfanilic Acid) ground to pass 80 mesh sieve and containing 16.75%S (as SO_3) by weight

(3) Ultimate coal standards (Leco Corp.); powdered coals with certified total sulfur and component sulfur (organic, pyritic, sulfatic) values

(4) Combustion crucibles (Leco Corp.);

(5) Scintillation vials (Fisher Scientific Ltd.); glass, 20 ml capacity vials with aluminum foil lined caps; vials kept sealed until required.

B. ANALYTICAL METHODS

The following description primarily applies to those specialized procedures utilized for data acquisition and processing of chronograms generated from coal samples. As such, the correct applications of these ancillary techniques requires a thorough understanding of the general operational parameters provided by the Leco instruction manual.

1. System Calibration

On a daily basis or after any system modification or maintenance a single point sulfur calibration should be performed. At the combustion temperatures required for chronogram generation (550°C or lower) only ABSA or non-sulfate ultimate coal standards can be reliably used. After calibration is completed, a system blank should be determined using a combustion boat containing about 1 g of powdered silica. To ensure the lowest possible blank, it is recommended that previously heat treated (1370°C for 3 minutes under 0_2 flow) silica and crucibles be utilized.

ABSA sulfur standards must be prepared using heat-treated silica to ensure that the sulfur released is derived only from the ABSA-SO_3 group. Using an analytical balance and scintillation vials various amounts of ABSA are added to appropriate weights of silica and the mixture well shaken to yield a uniform suspension. It is critical to the reproducibility of the results that the mixture be shaken enough. It is recommended that a laboratory mixer of the type used to agitate test tubes be used. An illustrative example of preparation of several such standards is given below:

Standard No.	ABSA (mg)	Silica (mg)	Sulfur (mg)	Sulfur (%)
1	6	994	1	0.1
2	30	970	5	0.5
3	60	940	10	1.0
4	90	910	15	1.5
5	120	880	20	2.0
6	180	820	30	3.0
7	240	760	40	4.0
8	300	700	50	5.0

Having selected the desired combustion temperature and run several (i.e., 3 to 5) analyses to condition the Leco instrument, perform the following procedure using an appropriate sulfur standard (containing no sulfate):

(1) Invoke the ACQFIX program on the Compaq and weigh a convenient amount of sulfur standard (usually 100 mg) into crucible; enter the weight and %S of the standard.

(2) Place Leco into load furnace mode, start ACQFIX data acquisition, immediately slide crucible into furnace, and place Leco into the analysis mode; if during analysis the observed voltage output exceeds the 5V data acquisition range, terminate analysis and repeat steps 1 and 2 using about 1/2 standard weight used previously; if voltage output is within the voltage range proceed to step 3.

(3) The computer will automatically stop data acquisition when 300 points have been taken. Repeat steps 1 through 3 until between 3 and 5 replicate standard analyses have been run. Run the program LINEAR for each of the acquired files to generate linearized chronograms.

(4) Run the program CAL using the linearized files generated in step 3. The CAL program will calculate an average calibration factor and store it in the file \DATA\ MULT.FAC.

(5) Select ACQFIX data acquisition program and weigh and enter a convenient amount of an appropriate sulfur standard.

(6) Repeat steps 2 and 3; if the observed %S and expected %S of the standard are within acceptable limits of variation, proceed with samlple analyses; if variability is not acceptable several of the routine maintenance procedures (refer to appropriate section) may be required.

(7) Repeat entire system calibration procedure after any required maintenance.

2. Sample Chronogram Analysis

At reduced furnace combustion temperature (i.e. 550°C), a 6-8 minute analysis time is required for chronogram generation. The minimum Leco data acquisition time per sample should therefore be routinely set at 4 minutes. To save time, vials containing 900 (±2%) and 450 (±2%) silica (preferably heat treated) may be preweighed, capped, and stored until needed. To acquire chronograms, the basic procedure listed below must be followed. It must be noted that reproducability of the spectrochronograms depend very much on how uniformly sample has been redistributed in SiO_2 and afterward in a boat.

(1) Select ACQFREE data acquisition program (with updated calibration factor); switch lance gas off on Leco.

(2) Weigh 100 mg (±5%) of well shaken coal sample into vial containing 900 mg (±2%) silica such that a 9:1 silica-coal ratio is obtained.

(3) Enter exact weight of coal and sample identification into the ACQFREE program.

(4) Shake silica-coal mixture vigorously to ensure a uniform suspension; quantitatively transfer entire suspension evenly across bottom of combustion crucible (preferably heat-treated); tap boat several times to pack suspension into boat.

(5) Place Leco into load furnace mode, start ACQFREE data acquisition program, wait for "load furnace" LED to come on, then immediately slide crucible into combustion furnace, and put Leco into the analysis mode by pressing "ANALYZE".

(6) If the observed voltage exceeds the 5V data acquisition range, terminate the analytical run and repeat steps (2) through (5) using one half the silica and coal weights previously used (i.e., reduce silica to 450 mg and coal to 50 mg).

(7) If the voltage output remains within acceptable limits stop computer data acquisition immediately upon completion of the Leco sample analysis by pressing the ESC key. Run a duplicate analysis under identical conditions to determine chronogram reproducibility.

(8) For each file that was acquired in steps 5 through 7 (one for each analysis), run the program ADJUST. This will generate an "adjusted" file whose filename has the prefix "A". Run the program DECON for each adjusted spectrochronogram.

Chronogram reproducibility will depend to a great extent on the degree of consistency obtained when preparing the silica-coal mixtures. Since a silica to coal ratio of 9:1 is routinely used, the silica should be as free of sulfur as possible. A simple expedient to remove contaminant is to fill crucible with approximately 3 g of silica and heat at 1370°C for about 3 minutes. Once cooled, the heat-treated silica should be immediately transferred to a sealed container to prevent atmospheric moisture uptake thereby preventing silica particle agglomeration. Coal samples should be dry and of the finest particle size possible to ensure maximum uniformity in the silica:coal suspsension.

3. Leco Furnace Maintenance

Continuous operation of the Leco furnace at lower than recommended temperature (i.e., 550°C) requires an upgraded maintenance program over that specified in the Leco manual.

Incomplete combustion of higher molecular weight sample organics at reduced furnace temperature gradually fouls combustion gas delivery and detection components which is eventually expressed as increasing variability in the sample chronograms. Therefore, if sample preparation is consistent but chronogram irreproducibility is still high, the following steps should be taken:

(1) Check combustion gas flows and IR cell voltage. If the flow is incorrect the mass flow controller may be fouled and may require cleaning. If IR cell voltage has dropped below 7.7V adjust upwards to 8.5v.

(2) Change both anhydrone drying tubes. Run several conditioning samples (i.e. 2 to 4) followed by replicate chronogram samples. If chronogram reproducibility is still poor (i.e. greater than 10% RSD) proceed to step 3.

(3) Increase furnace temperature to 1370°C, allow furnace to equilibrate and run several conditioning samples. At this time heat treated silica and crucibles may be prepared if desired. Reduce temperature to 550°C by stages (i.e. 1370 to 900 to 550°C). Run several conditioning analyses followed by replicate chronogram samples. If variability still high proceed to step 4.

(4) Remove delivery tube assembly from collection/combustion tube assembly and anhydrone filter. Thoroughly clean and dry delivery tube line and anhydrone filter teflon connector. Remove top plug from combustion/collection tube assembly metal plate and clean internally with metal rod. Re-insert top plug and reconnect delivery tube assembly. Run several conditioning analyses followed by replicate sample chronograms; if high irreproducibility proceed to step 5.

(5) Switch Leco furnace off and allow instrument to cool to room temperature. Disconnect delivery tube assembly, purging tube, and lance connector tube (clean all thoroughly). Remove furnace liner with half boat stops and combustion/collection tube assembly. Check both components for fractures; if fractured replace. If no fractures clean all components using compressed air. Re-insert and reconnect all components. Remove top and side panels from Leco and inspect all plastic connecting lines and Balston filter. Replace any highly contaminated lines and replace filter. Conduct leak check as by procedures given by Leco manual. Replace Leco panels and switch Leco furnace on to desired temperature.

BARRINGER

5581D/0471A

CONTRACT REPORT NUMBER 59219-01-SQ

REQUISITION NUMBER 23440-5-9219

CONTRACTOR: BARRINGER RESEARCH LIMITED

TITLE: CANMET SULFUR-FORM ANALYZER
FINAL REPORT

FINAL REPORT
FOR THE DEVELOPMENT OF A CANMET SULFUR-FORM ANALYZER

DSS File No. 06SQ.23440-5-9219

Prepared for:
Department of Supply and Services
Science & Professional Services Directorate
11C1 Place du Portage, 11 Laurier Street
Hull, Quebec
K1A 0G1

Prepared by:
Barringer Research Limited
304 Carlingview Drive
Rexdale, Ontario
M9W 5G2

166 pp

BRL File No. 301.11

February, 1989

EXECUTIVE SUMMARY

Methodology related to sulfur form analysis in coal by differential thermal analysis has been investigated. A commercially available total sulfur analyzer was modified to measure thermal decomposition/oxidation of sulfur containing compounds in coal as a function of time. Variables associated with the kinetics of oxidation and dissociation of the different forms of sulfur in coals and the instrumental parameters associated with combustion and analysis were investigated.

Initial studies conducted on synthetic coal prepared from charcoal and known amounts of different sulfur form containing compounds indicated that the local temperature rise was due to the initial coal combustion processes. The local increase in temperature resulted in poor reproducibility of the peak shape and the chronogram time for the individual sulfur forms. To improve resolution of the spectro-chronograms, a number of catalysts were investigated at various furnace temperatures. The use of these catalysts neither improved specificity nor reproducibility.

Peak resolution of the spectro-chronograms was observed to improve considerably upon variation of the furnace temperature, however, peak overlap and reproducibility was still not sufficient for quantitative analysis using peak deconvolution. The addition of silica oxide as a neutral filler was found to give complete separation between sulfate and organic/pyritic SO_2 peaks with reproducibile spectro-chronograms.

Studies related to the analysis of real coal samples indicated substantial variations in the spectro-chronograms for different types of coals. To better control the oxidation/decomposition rate, coal samples were diluted with various solid matrices (i.e. silica, alumina, clay, salt, titanium dioxide, and calcium carbonate). Only silica was found to be inert with regard to sulfur release. Operating temperature and the silica to coal ratio were then optimized with respect to the spectro-chronogram reproducibility and sulfur form speciation.

Based on partial speciation and good spectro-chronogram reproducibility peak deconvolution technique was implemented for further quantitative identification of the different sulfur forms.

A microcomputer was attached to the sulfur analyzer to acquire spectro-chronograms and to deconvolute the overlapped peaks. Marquardt's algorithm was used to fit the observed spectro-chronograms to models containing two and three Lorentzian peaks. The spectro-chronograms usually showed an initial small peak due to one type of organic sulfur and then a larger peak which was due to a second type of organic sulfur and pyrite. The overlap of the second organic peak with the pyrite peak was severe enough that the three-peak model could not consistently resolve it. Another approach was taken in which the observed spectro-chronogram was resolved into only two peaks. The relative contributions of organic sulfur and pyritic sulfur to the second peak were then inferred from th ratio of organic sulfur in the first peak to pyritic sulfur in the second peak. There has been evidence published showing this ratio to be a constant for a given coal bed.

TABLE OF CONTENTS

SECTION PAGE

I. INTRODUCTION 1

II. BACKGROUND 2

III. ANALYTICAL PROTOCOL FOR SULFUR FORM SPECIATION 6
  III.1 INTRODUCTION 6
  III.2 SYNTHETIC COST STUDY 7
  III.3 EFFECT OF INCREASING SILICA TO COAL RATIO ON SULFUR SPECIATION 18
  III.4 EFFECT OF TITANIUM DIOXIDE AND OTHER COMMONLY USED OXIDANTS ON SULFUR SPECIATION 22
  III.5 CHARACTERIZATION OF ARGONNE COAL STANDARDS 30
  III.6 SPECTRO-CHRONOGRAM REPRODUCIBILITY 30

IV. PEAK DECONVOLUTION 41
  IV.1 COMPUTER SYSTEMS 41
  IV.2 COMPUTER MODELS 45
  IV.3 SYNTHETIC SPECTROCHRONOGRAMS 54
  IV.4 PYRITE STANDARD ADDITION STUDY 67
  IV.5 APPLICATION OF CONSTRAINTS TO COMPUTER MODELS 68
  IV.6 THE K-FACTOR METHOD 74
  IV.7 LOCAL HEATER STUDIES 81
  IV.8 SYNTHETIC SPECTROCHRONOGRAMS-FURTHER STUDIES ... 83
  IV.9 COMPENSATION FOR DETECTOR NON-LINEARITY 110

V. DISCUSSION AND CONCLUSIONS 115

VI. REFERENCES 118

VII. APPENDIX: SOFTWARE LISTINGS 119

I. INTRODUCTION

There is a requirement to develop a simple sulfur form analyzer to be able to analyze sulfur compounds in coal. The present ASTM method is time consuming, complicated and requires highly trained personnel to perform the analysis for the three sulfur forms, namely: sulfate, pyritic and organic sulfur. Hence, research was initiated by CANMET to develop analytical methodology based upon differential thermal analysis.

This report describes the research conducted by Barringer Research Limited under DSS contract 06SQ.23440-5-9219 "The Development of a CANMET Sulfur-Form Analyzer". The program was contracted as a three phase program which providse a tested prototype instrument to CANMET for further evaluation. Phase I involved research studies related to the variables controlling the kinetics of oxidation and dissociation of different forms of sulfur in coal and the measurement of the effect of these variables. The Phase II program was associated with developing an analytical protocol for sulfur form analysis with existing commercial instrumentation. This involved experiments to define the optimum analytical parameters and the development of a microcomputer-based data acquisition and peak deconvolution system. The Phase III part of the program involved modification of the instrumentation according to the recommendations resulting from the Phase II study.

II. BACKGROUND

The three sulfur forms recognized by ASTM D2492 are: (1) sulfate sulfur which may be in the form of calcium or iron sulfate; (2) pyritic sulfur which is sulfur combined with iron in the forms of minerals, pyrite and/or marcasite; and (3) organic sulfur which is bonded to the carbon structure. For the evaluation of chemical and cleaning processes, it is important to determine various sulfur forms before and after cleaning. Treating coal with chemicals may change the mineral components of the coal and their solubilities in hydrochloric and nitric acids and hence the use of ASTM methods for cleaned coals may be unreliable (Maijgren et al, 1983). In this method sulfate sulfur is extracted from coal with dilute hydrochloric acid, precipitated as barium sulfate, ignited and weighed. After sulfate removal, pyritic sulfur is extracted with nitric acid and the extract iron is measured by Redox titration. Organic sulfur is calculated by subtracting sulfate and pyritic sulfur from total sulfur. This method has been found to be labour intensive and unable to provide rapid turnaround of sample analysis.

Preliminary results reported by Moran and Sirois (1984) on "Method for Analyzing Different Sulfur Forms" in coal indicated that thermal decomposition/oxidation of sulfur forms from coal appeared promising, however, further investigation was required. Results from the study indicated that temperature in the combustion chamber can change the peak position and the characteristics of the spectro-chronograms. The sample mass and the method of placing the sample on the boat also affected the spectro-chronograms. In addition, large discrepancies between spectro-chronograms of pure sulfates and sulfates in coal were noted. Therefore, a request for proposal was initiated by CANMET, Energy, Mines and Resourcs and Barringer

Research was contracted to perform further research and development on the methodology, to develop the analytical protocol for sulfur form analysis of coal and to modify and deliver instrumentation to CANMET for routine analysis of the sulfur forms in coal.

Phase I of the program was directed towards investigation of the thermal decomposition/oxidation of the sulfur forms from coal using existing commercial instrumentation with minor modifications. The primary objective was to develop a procedure which resulted in spectro-chronograms with reproducible peak shapes and chronogram times and maximum resolution between the different sulfur forms present in coal. Achieving this goal would dramatically simplify the methods of data reduction for quantitative analysis of the sulfur forms in coal.

A commercially available total sulfur analyzer (Leco Instruments Limited) was purchased and modified to detect sulfur oxide products from coal as a function of time during heating. This measurement provided a real-time spectro-chronogram plotted on a chart recorder.

Initial research addressed both the variables associated with the kinetics of oxidation and dissociation of the different forms of sulfur in coals as well as the instrumental parameters associated with combustion and analysis. In addition to the kinetics study, investigation of the heat transfer in coal during the initial combustion process was also addressed. In particular, the local temperature rise and its effect on the observed spectro-chronograms of pyrites and sulfates in coal was investigated.

Instrumental parameters such as furnace temperature, presence of lance gas, particle size and sample distribution within the reaction boat were also studied with synthetic coals consisting of charcoal and individual compounds or mixtures of the sulfur forms. These studies resulted in the development of an analytical method of good chronogram reproducibility and partial selectivity so that quantitative analysis using peak deconvolution processing techniques appeared feasible.

A 16-bit microcomputer was purchased along with hardware and software to provide a data acquisition capability. In the beginning of the computer work on this project, data was acquired on the microcomputer and then transferred to a larger computer. The larger computer already had software in place that could readily be modified to do peak deconvolution. In this way, the feasibility of the deconvolution software could be most quickly and easily established. Once this feasibility was established, a FORTRAN compiler was purchased for the microcomputer and all the necessary software was transferred to the microcomputer. The microcomputer thus formed a completely self-contained system to acquire data and do the deconvolution to provide final results.

Investigations were carried out using models that had two and three Lorentzian peaks. The task of the deconvolution software was to find the best fit of the observed chronograms to the given model. Software was written to generate synthetic spectrochronograms for the purpose of testing the deconvolution programs.

Software was written to allow daily calibration. Software is also in place to compensate for the non-linear response of the infrared detector.

In the course of this project, it became apparent that the overlap of the pyrite peak with a peak due to one form of organic sulfur was severe enough to make a deconvolution into three peaks unreliable. The software was modified to use an alternative approach (Moon, 1987; Yurovskii, 1974). This approach uses the observation that the ratio of organic sulfur in the first peak to pyrite sulfur is constant for a given coal deposit.

III. ANALYTICAL PROTOCOL FOR SULFUR FORM SPECIATION

III.1 INTRODUCTION

Initial studies of sulfur form speciation were conducted with synthetic coals and these studies were then followed with real coal samples analyses and sulfur form speciation.

To better control the oxidative/decomposition rate, coal samples were diluted with various solid matrices. From the various solid diluents tested (i.e. silica, alumina, clay, salt, titanium dioxide, and carbonate) only silica was found to be inert with regards to sulfur release. The other materials were found to either repress pyritic and organic sulfur decomposition or to speed-up decomposition/oxidation processes for all sulfur forms indiscriminantly.

The rather dramatic effect of silica addition with regards to chronogram reproducibility and sulfur speciation for various Leco coals was observed. Addition of silica not only improved reproducibility but also enhanced a partial speciation of sulfur. Depending upon the coal type, either 1 or 2 minor component peaks were produced within the first three minutes followed by a major peak after about four minutes decomposition time. Coal amendment studies using benzene sulfonic acid, napthalene sulfonic acid, and pyrite indicated that peaks appearing before three minutes consist of the more volatile aliphatic and aromatic sulfurs (i.e. 1 ring structures) whereas the major peak occurring after three minutes consisted of a mixture of polyaromatic (i.e. 2 or more rings) and pyritic sulfur components.

Although reproducibility was increased with increasing temperature above 500°C, speciation decreased such that by 600°C, no functional separation of organic and pyritic sulfur was observed. A decomposition temperature range between 500 and 550°C was found to be optimal.

As with the temperature, increasing silica to coal ratios from 2:1 to 20:1 increased chronogram reproducibility, after the coal sample was uniformly mixed with the filler. However, no increased speciation in the major component peaks was observed, for the ratios of silica to coal above 10:1. Increasing silica above this amount generally resulted in the broadening of all the chronogram peaks.

III.2 SYNTHETIC COAL STUDY

Quantitative oxidation-reduction of SO_2 products was found to be independent of the reducing media, i.e. coal; however, the rate of decomposition of different sulfates expected to be present in coal was found to be quite different. Iron sulfates were observed to decompose most readily while calcium sulfate was found to be most difficult to decompose (Figure III.1).

A series of experiments were conducted at various furnace temperatures (1375 to 400°C) with five different sulfur containing compounds that are known to be present in coal. The compounds under study were: the organic sulfur containing compounds such as provieded by LECO Inc., ferrous sulfate, hydrous ferric sulfate, hydrous and anhydrous magnesium sulfate, and hydrous calcium sulfate. The compounds were studied individually and in mixtures as well as in the presence and absence of reducing media. Two different coal media were used: (1) a high calorific coal containing 0.60% S, and (2) activated charcoal containing 0.08% S. Results indicated that the temperature requirements for an effective decomposition/oxidation process for different sulfate compounds

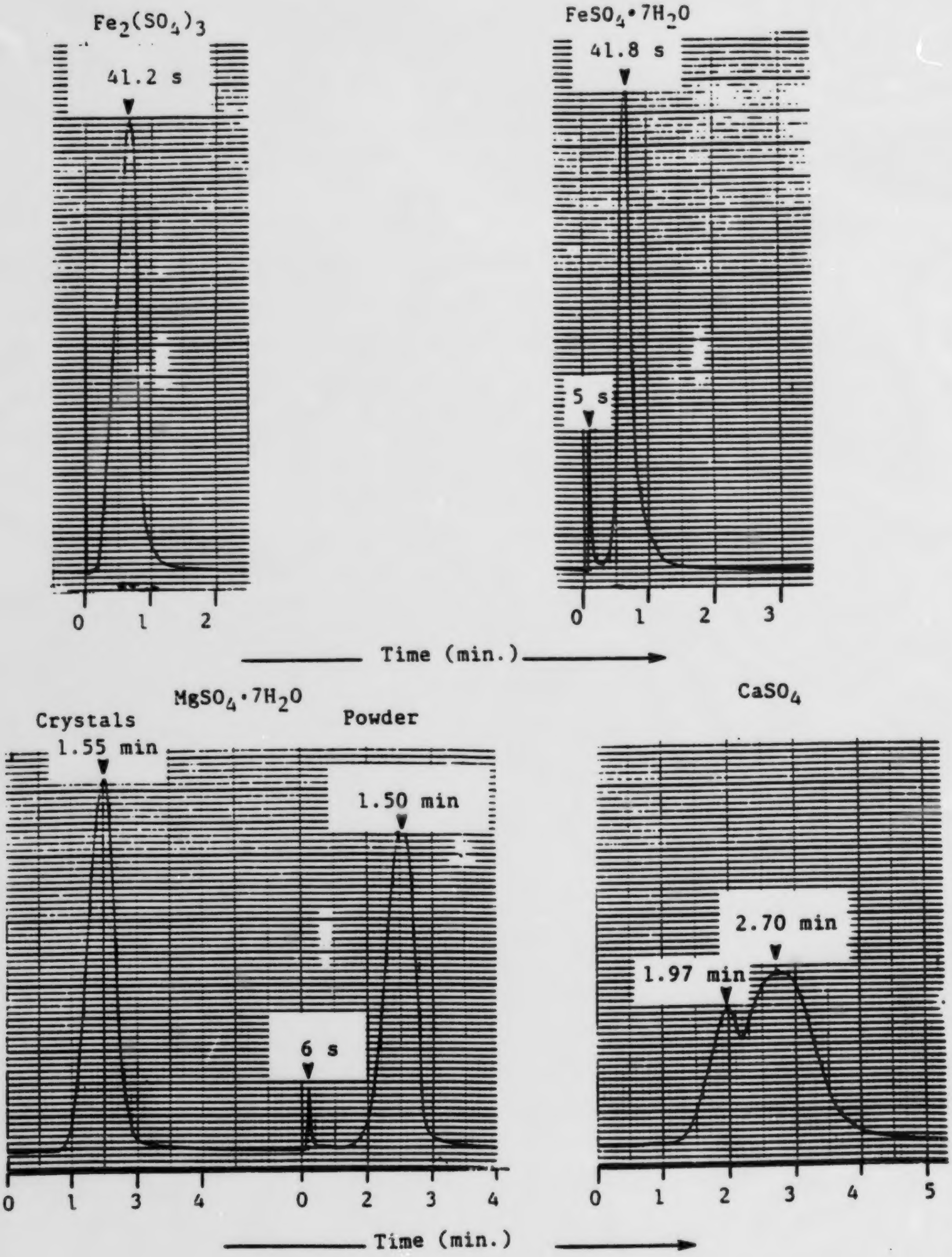


Figure III-1. Spectro-chronograms of sulfate analytes. Furnace temperature set at 1370 C with an oxygen flow rate at 3.25 L/min. and without lance gas.

were widely spread. At 1200°C, only partial oxidation/decomposition of $CaSO_4$ was observed with only 30% of the total sulfur reported while $MgSO_4$ resulted in about 70% recovery of sulfur. Addition of coal to the $CaSO_4$ at 1200°C had a marginal effect on the reported sulfur content, however, the addition of coal to $MgSO_4$ resulted in a total sulfur value. No differences in the sulfur content were observed with pyrite and the LECO organic compound since the temperature requirement for the oxidation of pyrite and organic compounds was below the set furnace temperature.

The presence of reducing media at 1375°C (i.e. coal) did not affect the percent of sulfur detected, however, the chronogram appearance was significantly different. Double and triple peaks for iron sulfide and sulfates were observed when the sample matrix was non-uniform. However, when care was taken to prepare a unifom sample, a single peak was observed. At lower temperatures, the presence of coal was found to affect both the chronogram appearance and the sulfur content in calcium and magnesium sulfates. The chronogram and sulfur content were observed to be dependent upon calorific value and that was primarily due to an increase in the local temperature during the coal combustion and not due to a chemical interaction with the coal.

Initial studies on the effect of different furnace temperatures on the chronograms were performed by using the temperature gradient within the furnace itself. The measured temperature as a function of temperature probe position within the furnace for the furnace temperature set at 1375 is shown in Figure III-2. Figures III-3, III-4, and III-5 are chronograms of sulfur containing compounds at different compositions within the temperature gradient. The objective of this study was to determine the temperature of complete combustion and to determine whether different sulfur compounds can be separated by temperature gradient.

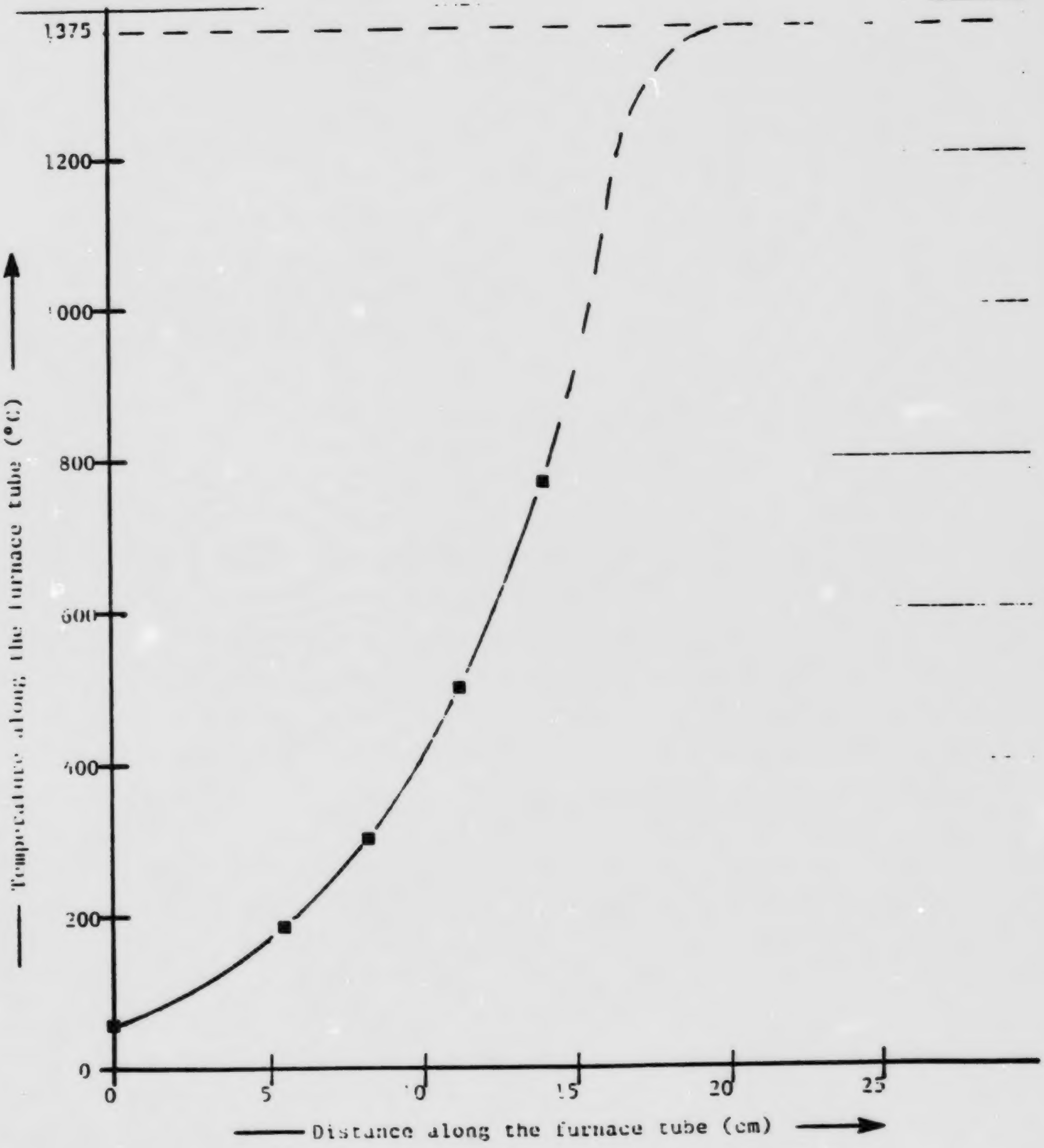


Figure III-2. Temperature gradient in the Leco furnace with the furnace temperature set at 1375 C.

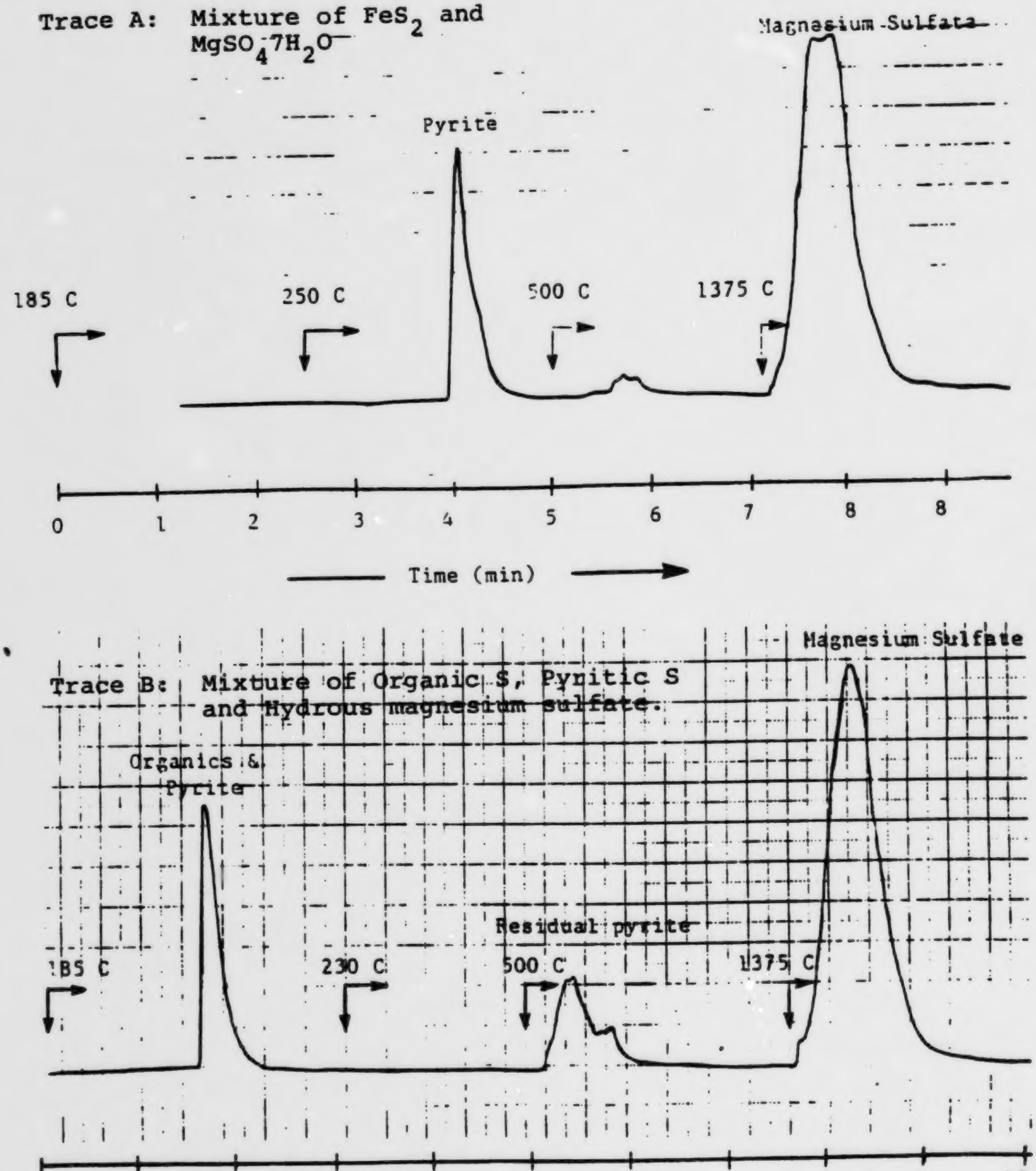


Figure III-3. Spectro-chronograms of sulfides and sulfates at various furnace tube temperatures. The second peak in both chronograms corresponds to the residual unreacted pyrite.

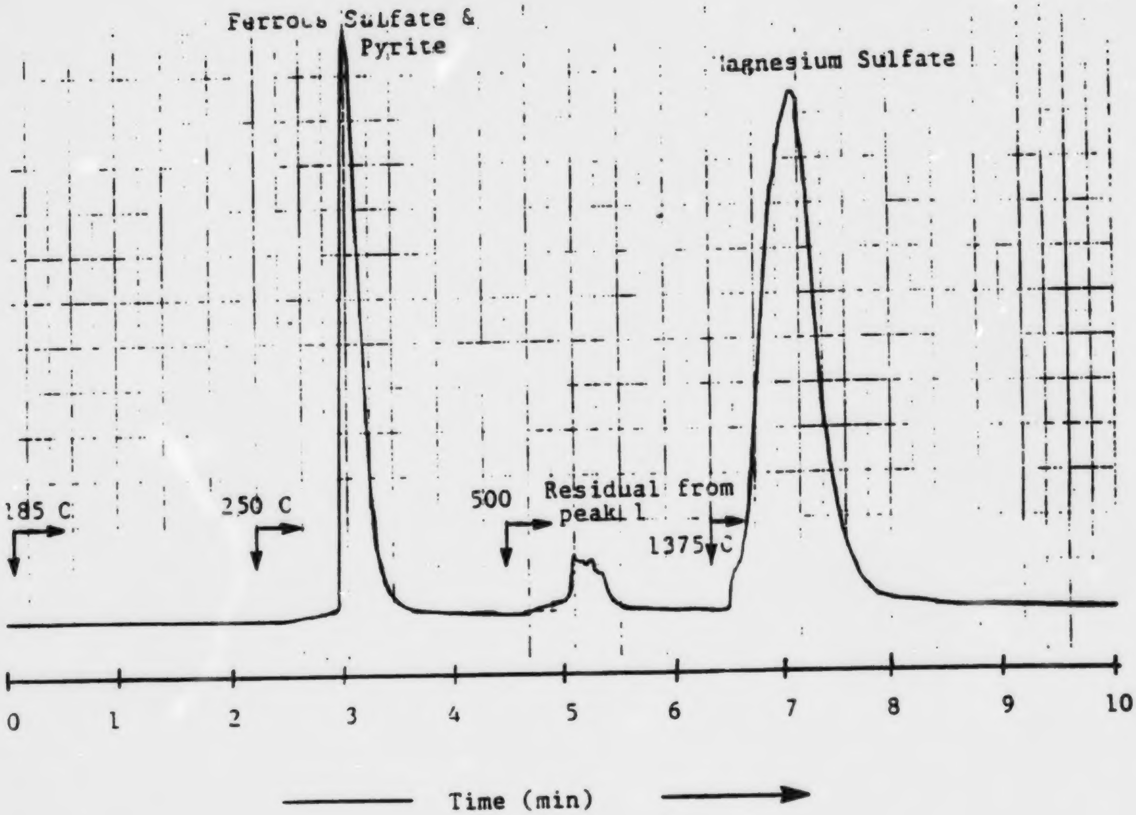


Figure III-4. Chronogram of pyrite, ferrous sulfate and magnesium sulfate in an activated charcoal (75μm). The sample boat was placed at four different positions within the furnace tube. The furnace temperature at those positions is indicated on the chronogram.

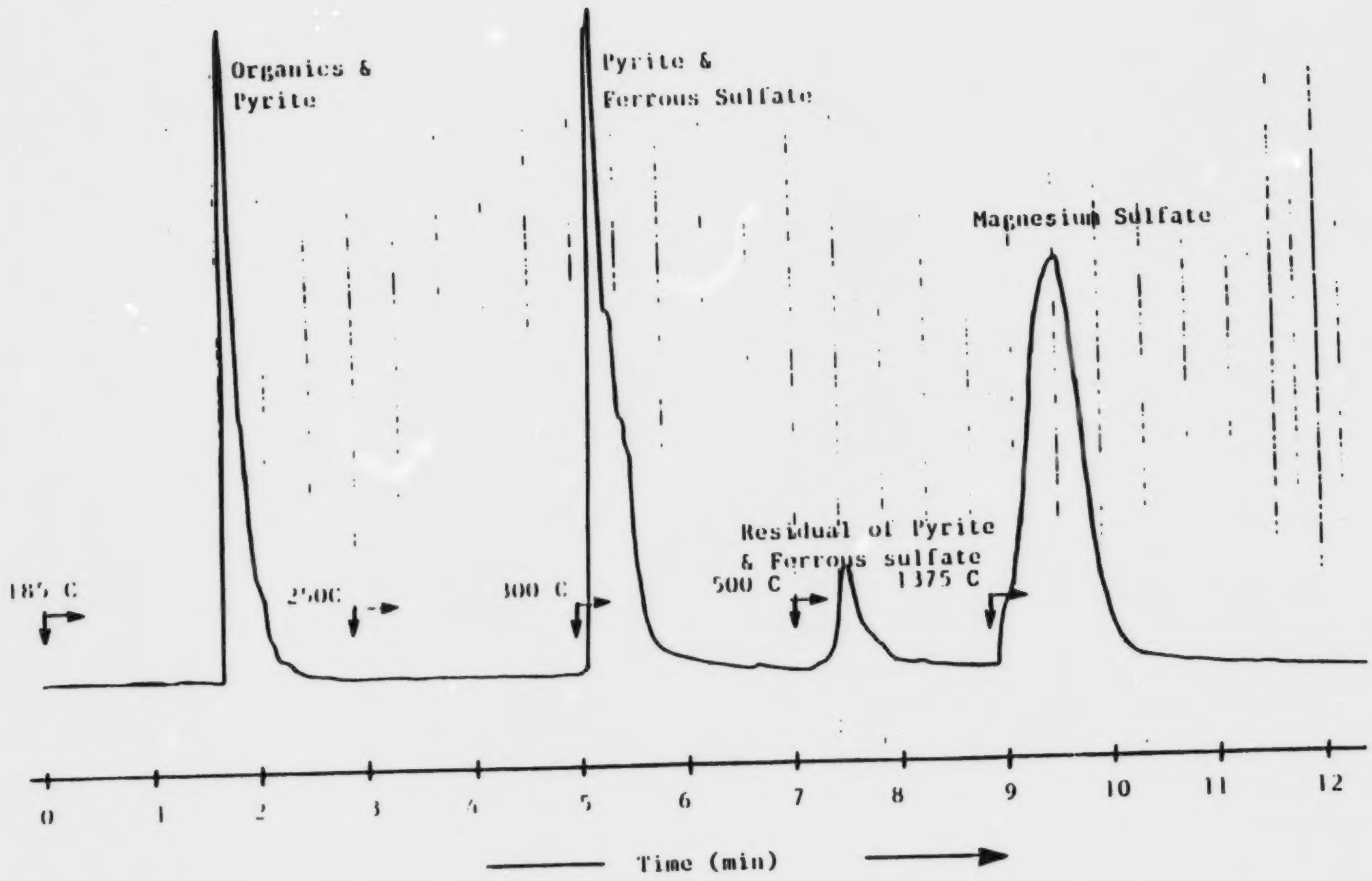


Figure III-5. Spectro-chronograms of organic, pyritic and sulfate sulfur in an activated charcoal ($\geqslant 75 \mu m$). The reaction boat was placed at five different positions within the furnace tube with corresponding temperatures indicated on the chronograms. The furnace temperature was set at 1375 C.

TABLE III-1

		All Five Compounds In	
$T^{o}C$*	Individual Compounds In Activated Charcoal	Activated Charcoal	Coal (0.66% S)
185	Organic	Organic (complete) FeS_2 (partial)	Organic, FeS_2, Some Iron Sulfates
305	FeS_2	FeS_2 $Fe_2(SO_4)_3$ Fe_2SO_4 } (partial)	$Fe_2(SO_4)_3$ $FeSO_4$
500	$Fe_2(SO_4)_3$ $FeSO_4$	$Fe_2(SO_4)_3$ $FeSO_4$ } completed	no further reaction
770	$MgSO_4$	$MgSO_4$ (partial) Some $CaSO_4$	$MgSO_4$ (partial) Some $CaSO_4$
1375	$CaSO_4$	$MgSO_4$ $CaSO_4$ } completed	$MgSO_4$ $CaSO_4$ } completed

---

* Furnace tube temperature at the sample boat position.

The position within the furnace for different compounds was established by mixing a known compound with activated charcoal and noting the minimum distance within the furnace at which at least 95% of the compound reacted. Upon completion of the initial reaction, the residual unreacted amount was determined by placing the boat at the end of the furnace tube (1375°C). Binary mixtures of compounds with similar temperature profiles were also tested, as well as a mixture of all six compounds.

As is seen from Table III-1 and Figures III-3, III-4 and III-5, individual compounds have sufficiently different temperatures at which they start to react, however, when the compounds are mixed with coal, there is no longer good separation of the compounds. The calorific value of coal has also a pronounced effect on the chronogram. Since the furnace temperature was far below the reaction temperature, at which decomposition and/or oxidation occurred for the compounds under investigation, it was decided to monitor the temperature during the reaction process.

For the first peak it was found that as decomposition/oxidation of the organic sulfur compounds began, the local temperature rose rapidly from 185°C to above 760°C and then slowly dropped down to near furnace temperature. Similar temperature changes occurred for the second peak.

To counteract the rapid rise in the local temperature, a neutral filler was added to the sample. Silica, titanium dioxide and several different clay samples were used for this purpose. Some interferences were found with clay fillers and titanium dioxide, however, the silica filler (60 um) was found to be chemically neutral and maintained a local temperature sufficiently low for the ferrous sulfate to remain unreacted at furnace temperature of 550°C. The results with and without silica filler are compared in Figures III-6 and III-7.

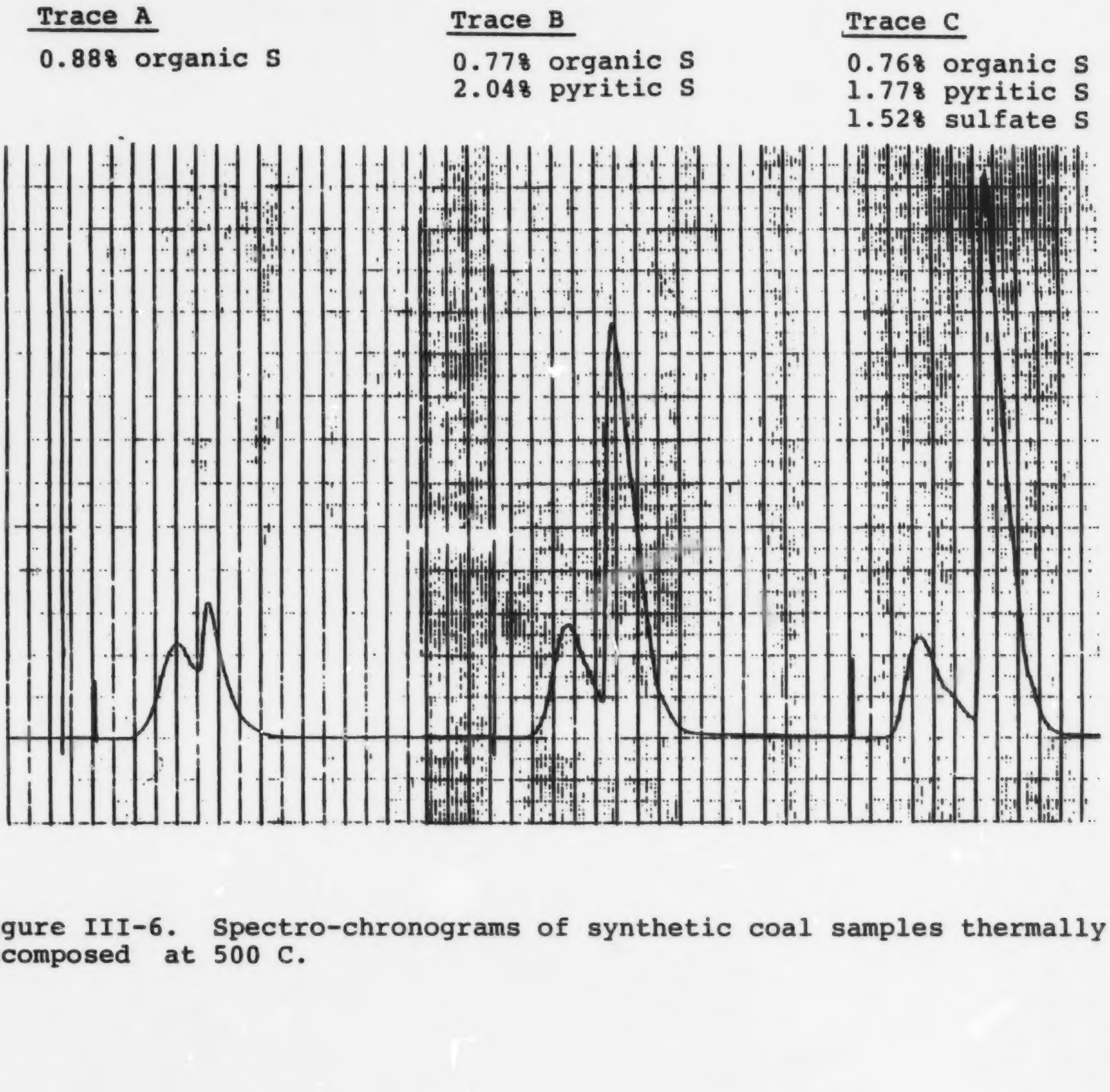


Figure III-6. Spectro-chronograms of synthetic coal samples thermally decomposed at 500 C.

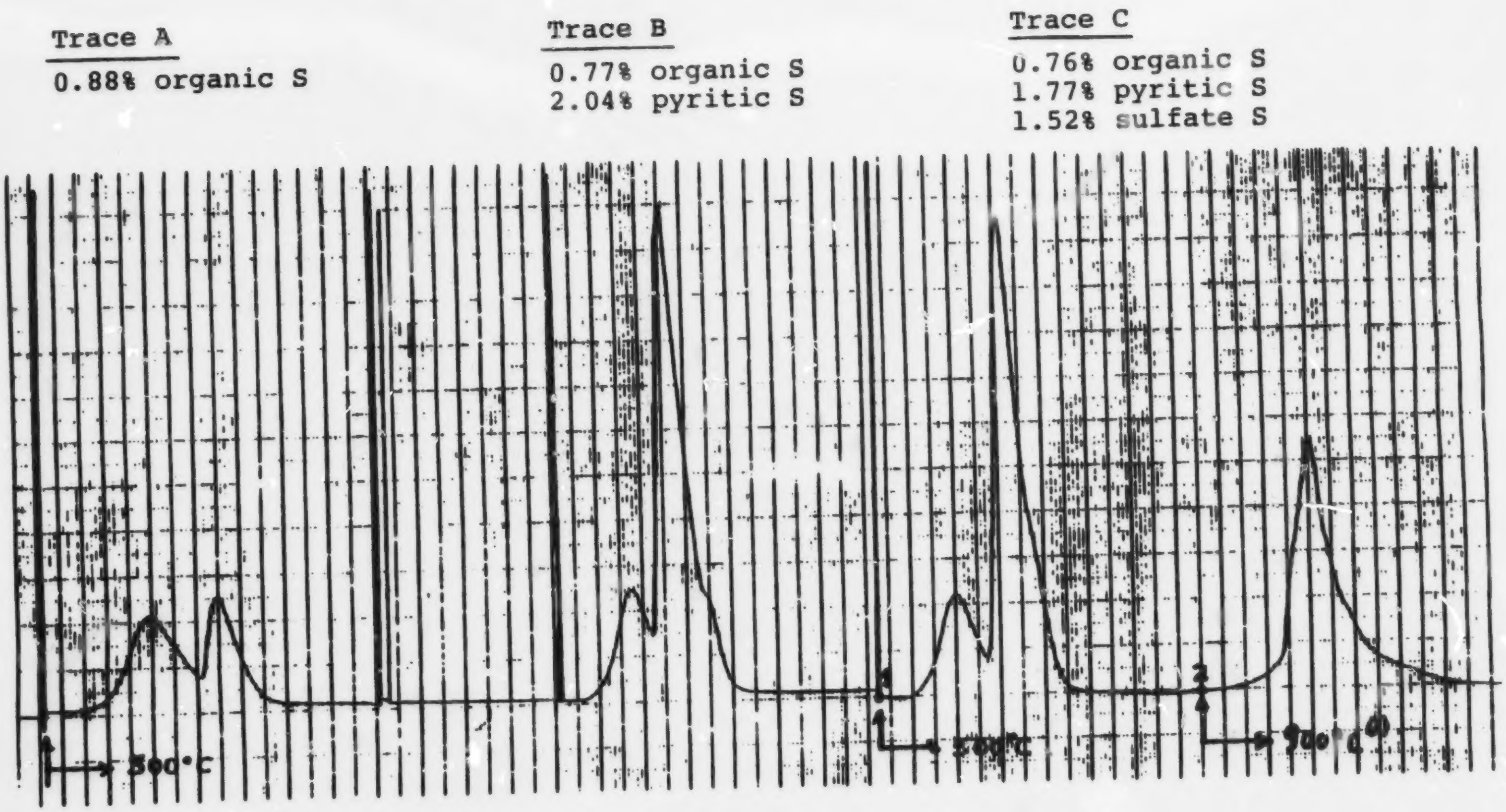


Figure III-7. Spectro-chronograms of synthetic coal mixed with silica in the ratio of ten to one.

Trace A: sample containing only organic sulfur compounds. Furnace temperature 500 C.
Trace B: sample containing organic and pyritic sulfur. Furnace temperature 500 C.
Trace C: sample containing organic and pyritic sulfur mixed with ferrous sulfate.
In Trace C the initial temperature was set at 500 C and upon completion of the first peak evolution the furnace temperature was set to 900 C without removing the sample boat from the furnace.

As can be seen from Figure III-6 iron sulfate cannot be separated from pyrite when silica was not added even at the furnace operating temperature of 500°C. However, by mixing coal with silica a good separation was obtained between the two peaks (see Figure III-7, Trace C).

To verify the uniqueness of silica and to establish the best ratio between silica and coal a whole range of coal samples have been investigated. Many of the samples were purchased from Leco Inc. and a few from Argonne National Laboratories, U.S.A.

III.3 EFFECT OF INCREASING SILICA TO COAL RATIO ON SULFUR SPECIATION

The coal sample containing only organic and pyritic sulfur (Leco coal 501-701) was used for this study. The ultimate analysis for this coal sample is given in Table III-2. The silica to coal ratio was varied from 2 to 16. The coal sample weight and furnace temperature were fixed at 100 mg and 550°C, respectively. The duplicate chronograms are displayed in Figures III-8 and III-9.

Table III-2

Summary of the Experimental Conditions and Results for the Various Silica/Coal Ratios at Furnace Temperature of 500 C.

Coal: L-501701

%S Organic = 0.71
%S Pyritic = 0.38
%S Sulfate = 0.02

Figure	Run	Coal mg	SiO_2 mg	Reported %S
III-8.1	A	102	203	1.08
	B	105	205	1.08
III-8.2	A	102	402	1.09
	B	102	403	1.10
III-8.3	A	101	601	1.04
	B	102	605	1.11
III-8.4	A	103	809	1.07
	B	104	802	1.10
III-9.1	A	105	919	1.11
	B	101	901	1.17
III-9.2	A	100	1015	1.14
	B	101	1010	1.10
III-9.3	A	100	1200	1.05
	B	104	1201	1.06
III-9.4	A	101	1601	1.07
	B	106	1609	1.11

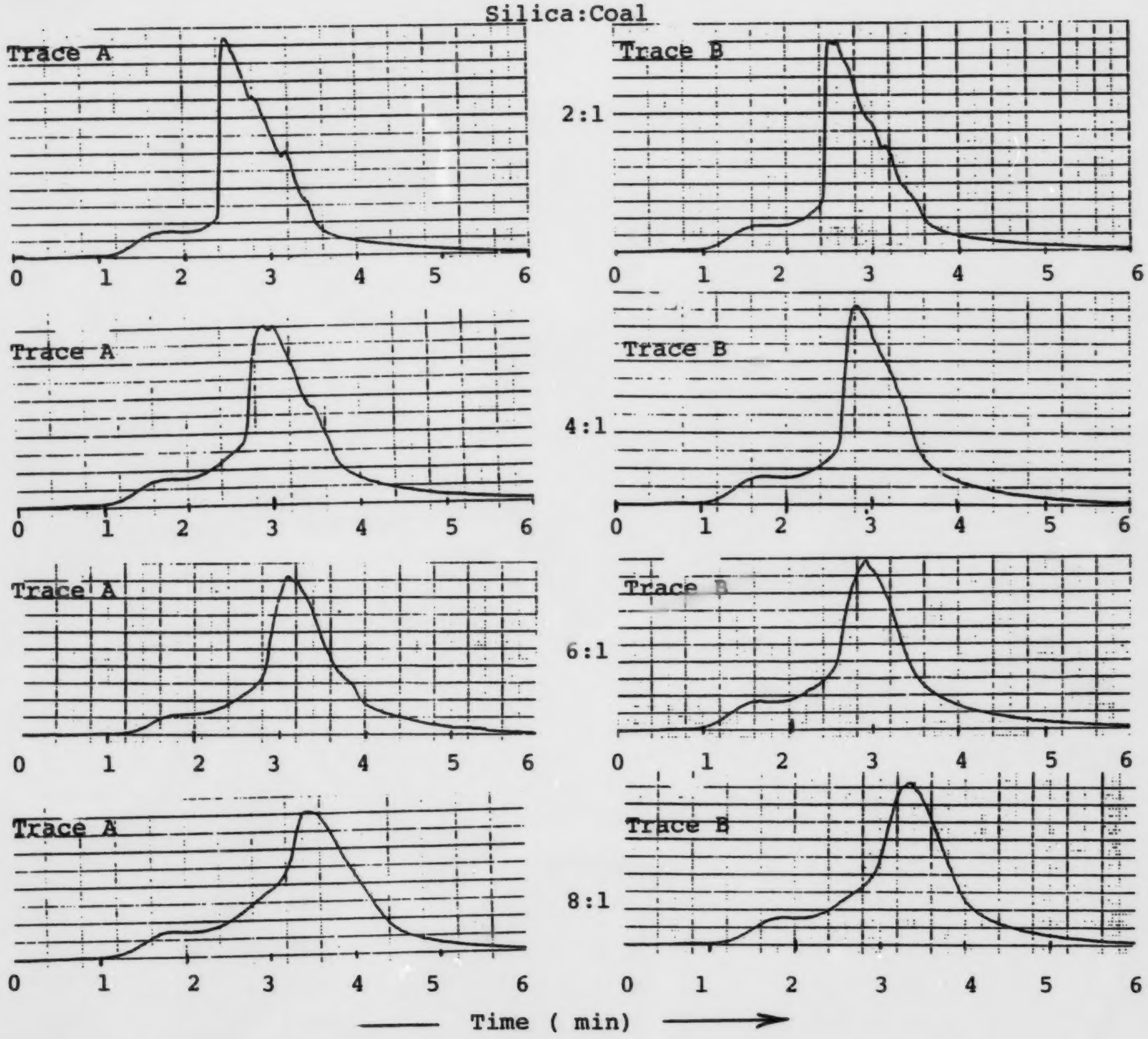


Figure III-8. Effect of increasing ratio of silica to coal on sulfur speciation at furnace temperature of 500 C. Coal: Leco 501-701.

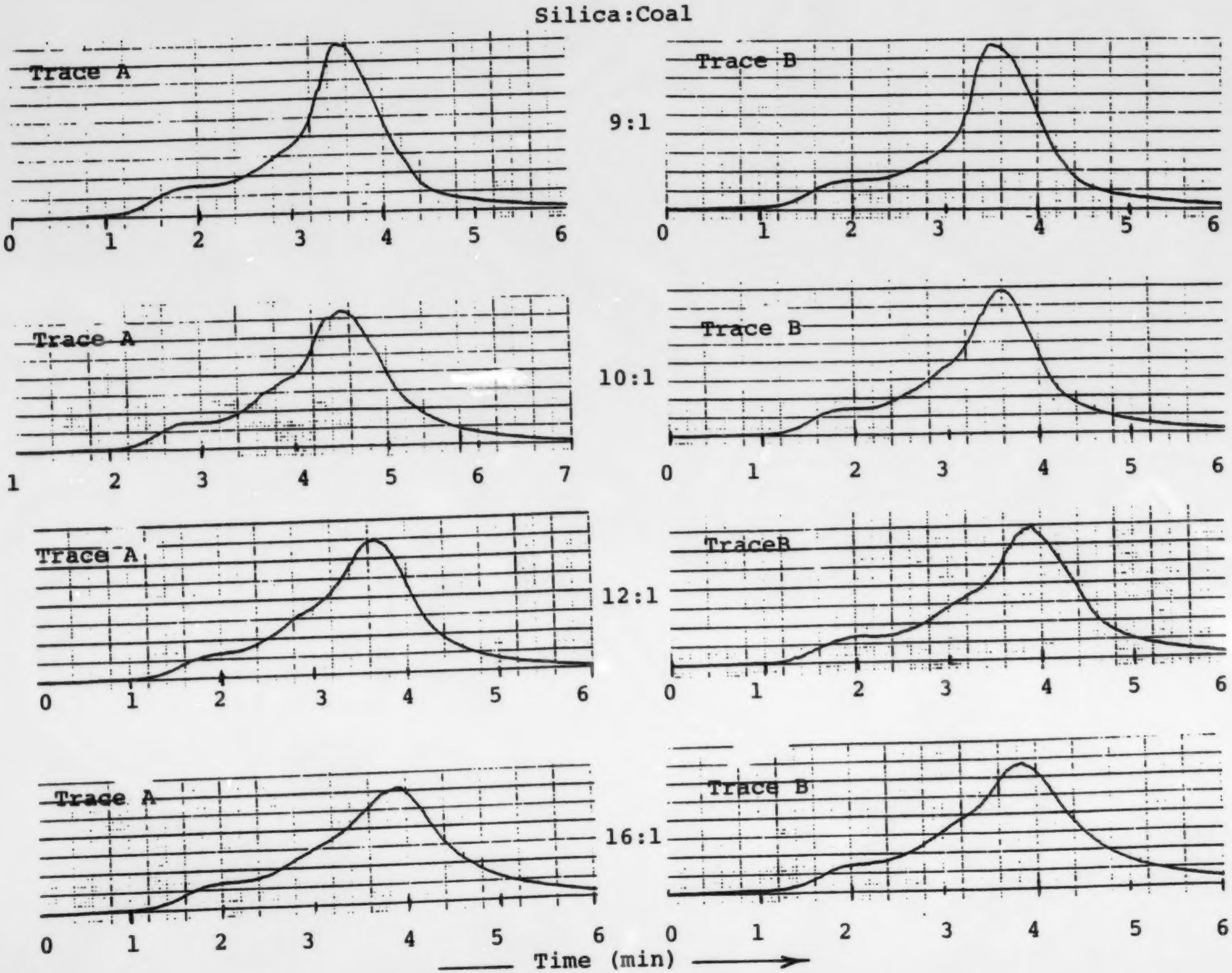


Figure III-9. Effect of increasing ratio of silica to coal on sulfur speciation at furnace temperature of 500 C. Coal: Leco 501-701.

Although peaks appear sharper at the low silica to coal ratio (Figure III-8) the reproducibility is not as good as at the higher ratio (8:1 up to 10:1) as shown in Figure III-9. The chronogram shape remains practically unchanged in that region of silica to coal ratio, thus the exact amount of silica is not critical in the range of 8:1 to 10:1. A higher ratio (greater than 12:1) progressively broadens the secondary peak with no increase in resolution between the two peaks as is seen in Figure III-9.

III.4 EFFECT OF TITANIUM DIOXIDE AND OTHER COMMONLY USED OXIDANTS ON SULFUR SPECIATION

A series of experiments have been carried out with TiO_2. The decomposition/oxidation of the sulfur containing compound was accelerated equally and no speciation was obtained. At 550°C and 9:1 ratio, titanium dioxide did not speciate sulfur as did silica (compare Figure III-12 with Figure III-9). Reducing temperature to 500 and 400°C did not result in improvement of speciation and only delay in sulfur peak appearance was observed. Lowering the ratio of titanium dioxide to coal down to 1:1 did not improve specificity (refer to Figure III-10). At low ratios (i.e., 1:1 to 2:1) multiple peaks were observed but chronograms were not reproducible, peaks were most likely spurious. At intermediate ratios (3:1 to 5:1) peak sharpening was observed along with loss in speciation and reproducibility. It thus appears that titanium dioxide has some catalytic effect on surlfur oxide formation. This property of titanium dioxide was then explored in the presence of silica, hoping that some added speciation may be attained.

Experiments were conducted with Leco coal #501-701 and furnace temperatures of 550°C. Chronograms from this study are displayed in Figures III-13 and III-14, with the experimental conditions summarized in Table III-4. Reduction in the degree of speciation was observed and reproducibility lost as titanium dioxide was increased.

Table III-3

Effect of Increasing Titanium Dioxide on Sulfur Speciation

Coal: L-501701
Furnace T.: 550°C
Total %S: 1.11

Figure No.	Run	Coal mg	TiO_2 mg	Reported %S
III-10.1	A	101	101	1.00
	B	101	101	1.10
III-10.2	A	106	200	1.14
	B	101	202	1.08
III-11.1	A	102	305	1.09
	B	102	303	1.08
III-11.2	A	107	403	1.07
	B	99	402	1.05
III-12.1	A	101	503	1.09
	B	101	503	1.09
III-12.2	A	105	701	0.99
	B	101	712	1.03
III-12.3	A	103	911	0.95
	B	103	916	0.96

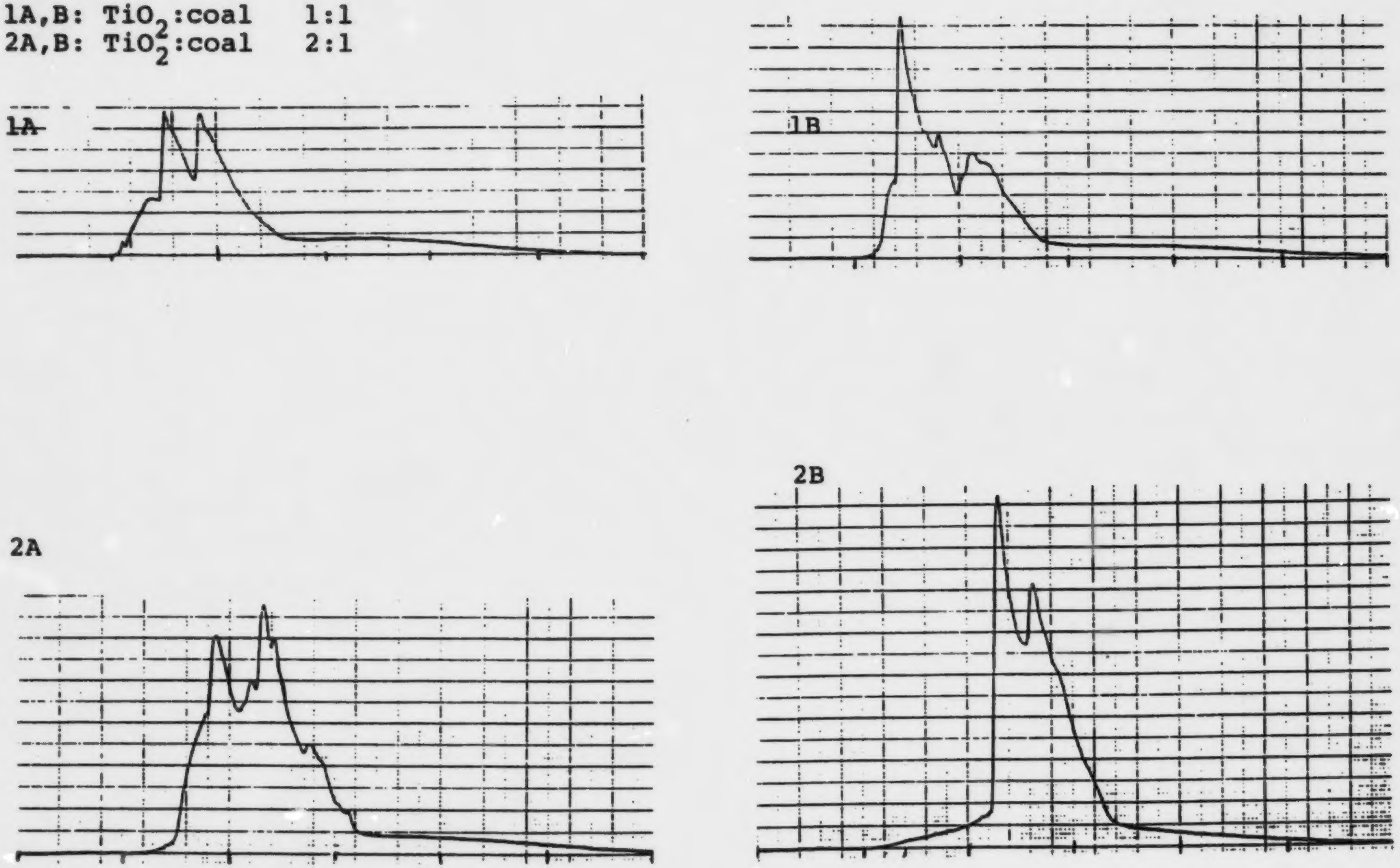


Figure III-10. Effect of increasing ratio of titanium dioxide to coal on sulfur speciation at furnace temperature of 550 C. Coal: Leco 501-701.

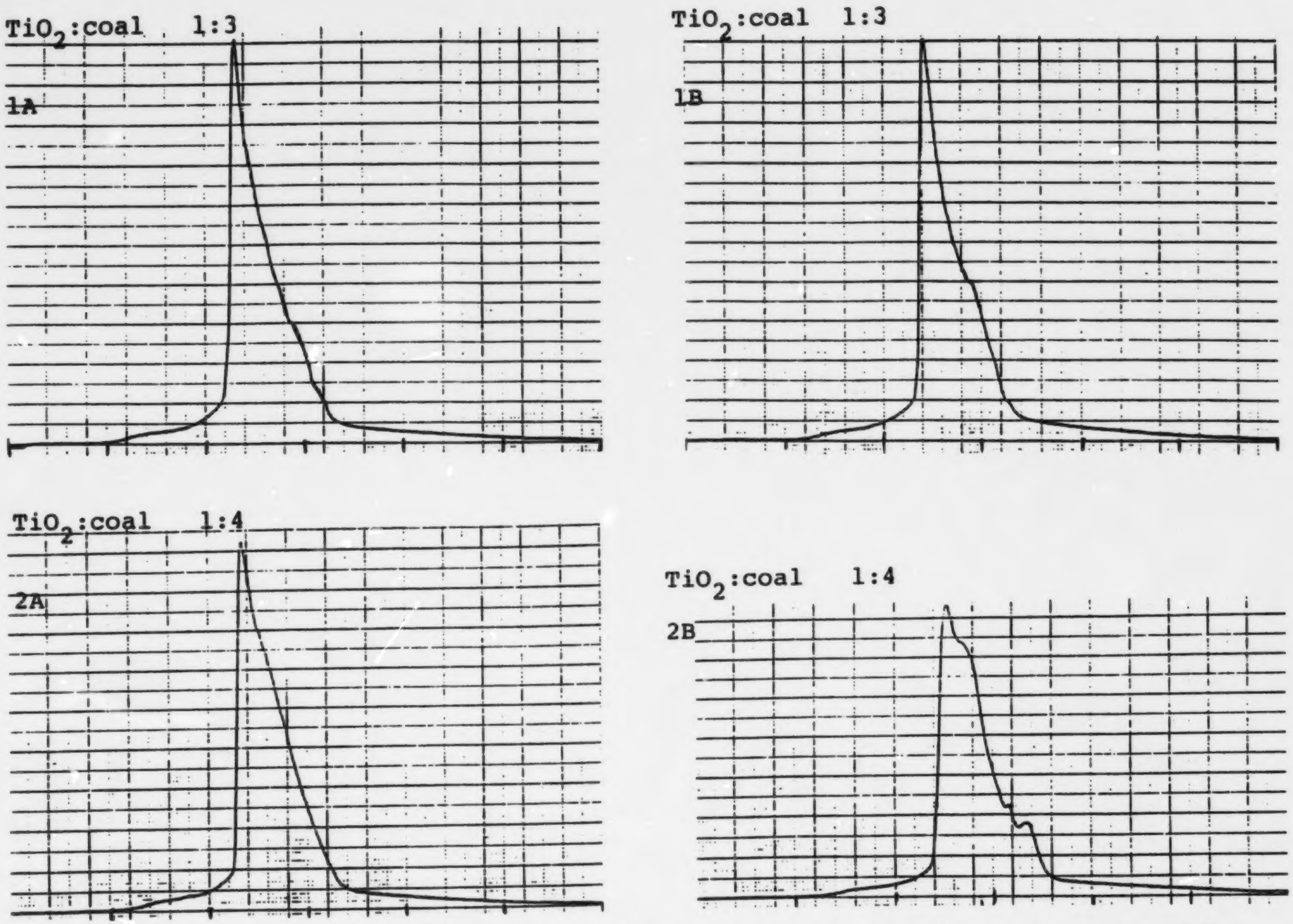


Figure III-11. Effect of increasing ratio of titanium dioxide to coal on sulfur speciation at furnace temperature of 550 C. Coal: Leco 501-701.

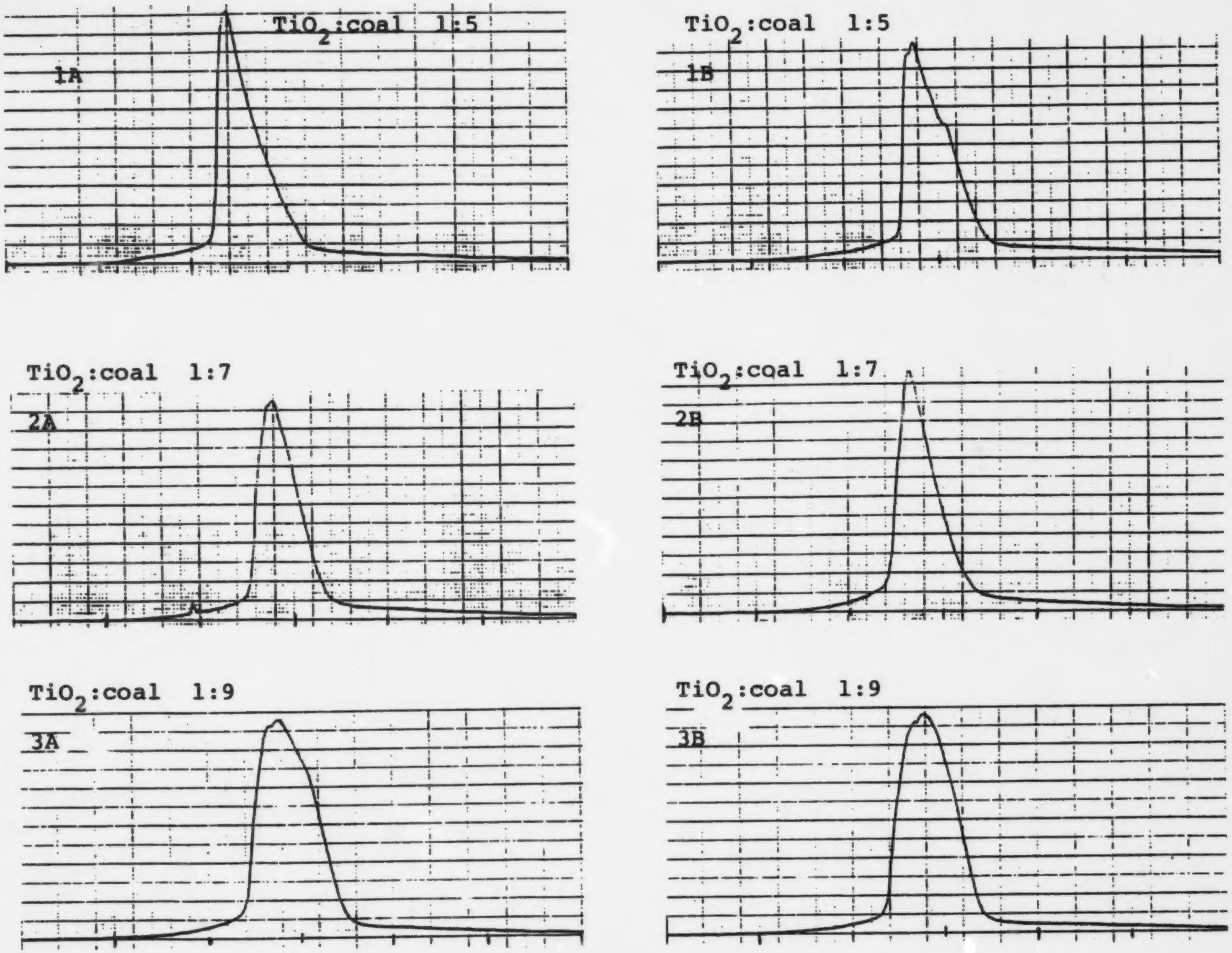


Figure III-12. Effect of increasing ratio of titanium dioxide to coal on sulfur speciation at furnace temperature of 550 C. Coal: Leco 501-701.

Table III-4

Effect of Titanium Dioxide on Sulfur Speciation in the Presence of Silica

Coal: L-501701
Furnace T.: 550°C
Total %: 1.11

Figure No.	Run	Coal mg	SiO_2 mg	TiO_2 mg	Reported %S
III-13.1	A	101	1010	0	1.11
	B	102	1010	0	1.12
III-13.2	A	100	905	101	1.10
	B	101	902	101	1.09
III-14.1	A	100	802	202	1.09
	B	103	801	200	1.07
III-14.2	A	102	600	400	1.04
	B	105	602	400	1.02

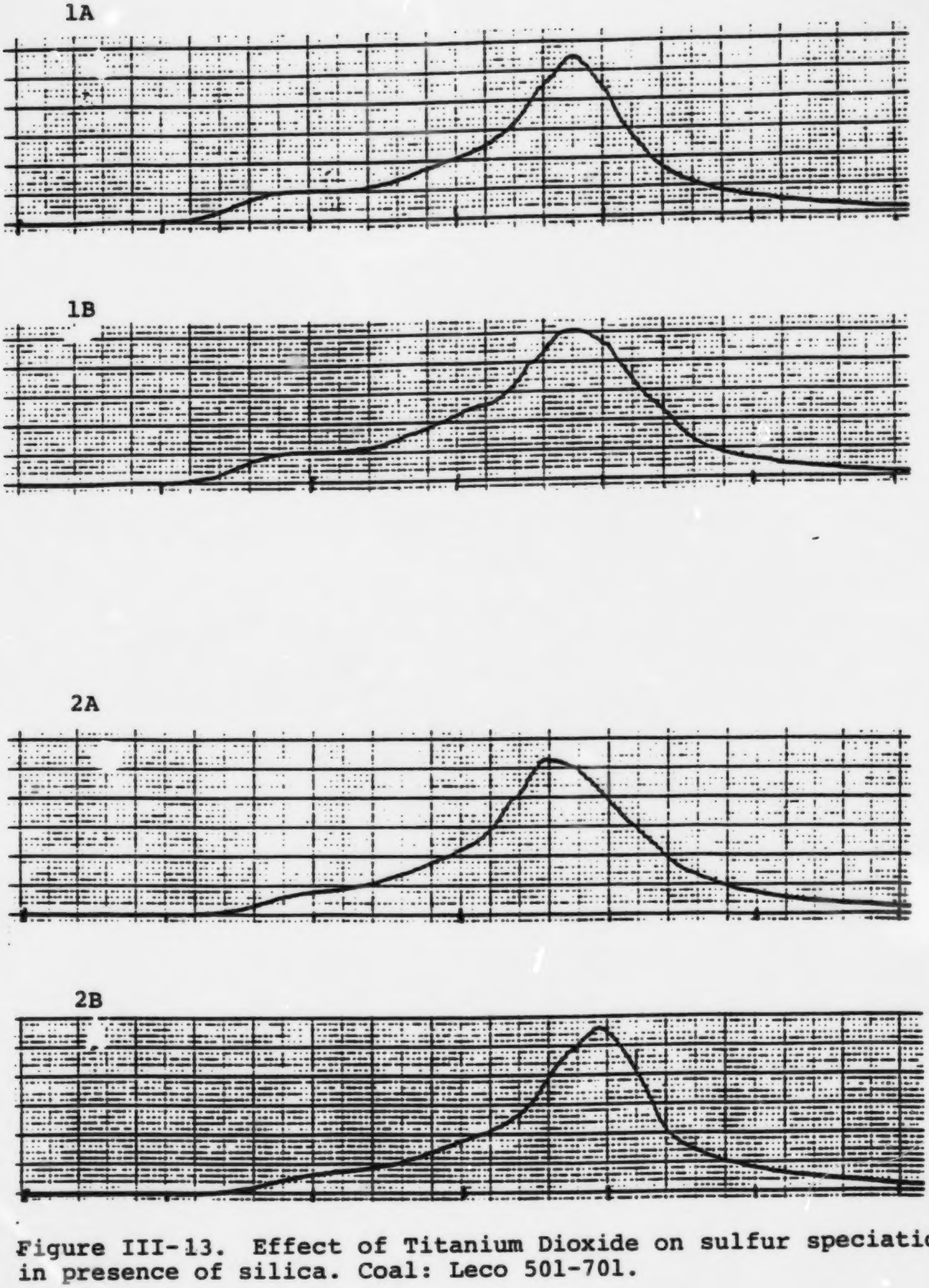


Figure III-13. Effect of Titanium Dioxide on sulfur speciation in presence of silica. Coal: Leco 501-701.

Trace 1A,B: TiO_2:coal 0:1; SiO_2:coal 10:1.
Trace 2A,B: TiO_2:coal 1:1; SiO_2:coal 9:1.

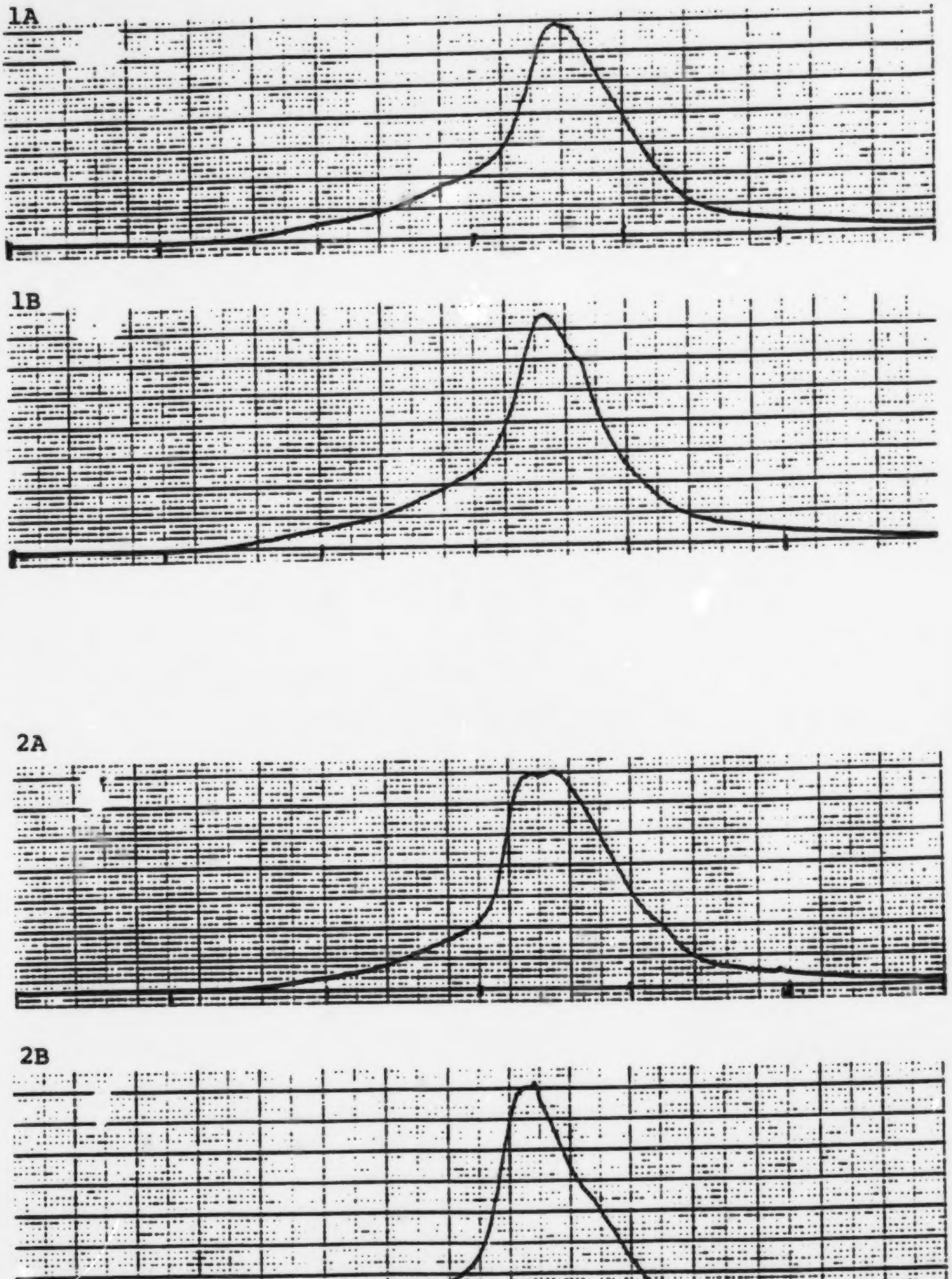


Figure III-14. Effect of Titanium Dioxide on sulfur speciation in presence of silica. Coal: Leco 501-701.

Traces 1A,B: TiO_2:coal 2:1; SiO_2:coal 8:1.
Traces 2A,B: TiO_2:coal 4:1; SiO_2:coal 6;1.

The addition of commonly used solid oxidants also did not yield a greater degree of sulfur speciation in the coals tested. Nitrates and permanganates were found to produce an explosive decomposition when added alone or in combination with silica. Other accelerators such as V_2O_5, Fe_3O_2, CuO and Ni_2O_3 did not increase sulfur speciation. In some cases (i.e. CuO, Ni_2O_5), sulfur release from coal was non-selectively inhibited. For these reasons, the further use of solid oxidants was discontinued.

III.5 CHARACTERIZATION OF ARGONNE COAL STANDARDS

In addition to Leco coal ultimate standards four standard coals were received from Argonne National Laboratory, Argonne, Illinois. All four standards (mesh 100) reported only a trace of sulfate sulfur. The coal samples were supplied in nitrogen filled sealed ampules. Some anomalies have been observed with analysis of these samples. Coal A202001 showed marked reduction in total %S at 550°C with or without silica (see Table III-5). This reduction in total %S suggests the presence of sulfate sulfur much higher than reported by the supplier. The sulfur speciation at 550°C with these coal samples is almost non-existent (Figures III-15 and III-16). Only one coal sample (A301004, Figure III-16.1) shows some indication of speciation.

III.6 SPECTRO-CHRONOGRAM REPRODUCIBILITY

For the deconvolution method to be applicable in addition to a partial separation obtained in the course of this study two other conditions need to be satisfied:

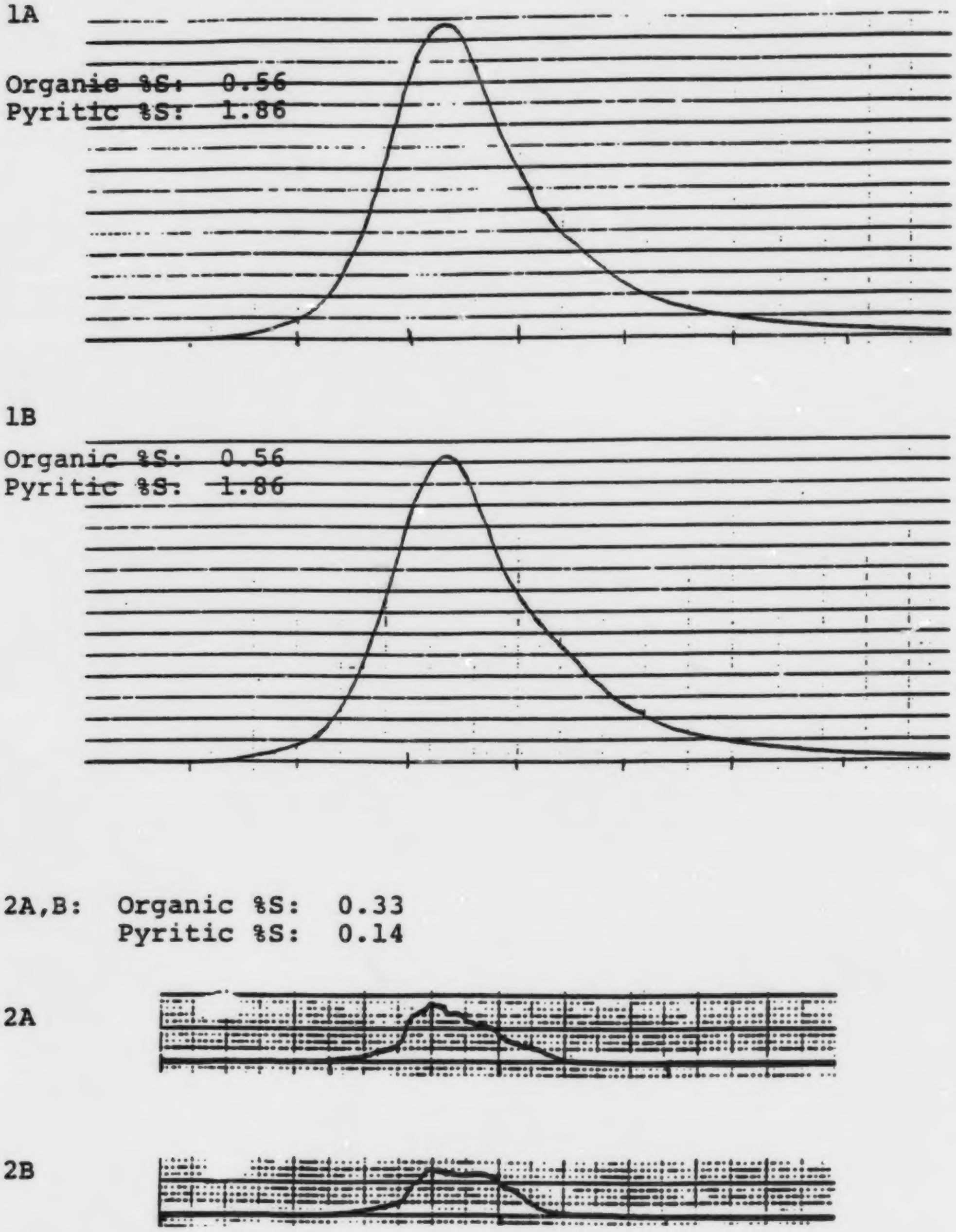
1. The spectro-chronogram have to be reproducible and
2. some additional information is required to establish the relationship between the organic SO_2 peak overlapped by pyritic SO_2 peak to that of the organic SO_2 peak preceeding the pyritic SO_2 peak.

Table III-5

Characterization of Argonne Coal Standards

	Suppliers Specifications for Sulfur Content				Observed Total S			
					Without Silica		With Silica	Figure
Coal ID	Organic S	Pyritic S	Sulfitic S	Total S	1370°C	550°C	550°C	No.
1. A101003	0.56	1.86	0.01	2.63	2.38 ± 0.02	2.44 ± 0.03	2.31 ± .05	III-15.1
2. A202001	0.33	0.14	0.05	0.52	0.97 ± 0.01	0.28 ± .01	0.10 ± .01	III-15.2
3. A301004	2.01	2.47	0.01	4.49	4.38 ± 0.05	4.43 ± .02	4.31 ± .03	III-16.1
4. A401003	0.71	1.32	0.01	2.14	2.25 ± 0.02	2.24 ± .03	2.28 ± .01	III-16.2

1. Upper Freeport medium volatile bituminous.
2. Wyodak sub-bituminous.
3. Illinois #6 high volatile bituminous.
4. Pittsburgh #8 high volatile bituminous.



1A: Organic %S: 0.56
Pyritic %S: 1.86

1B: Organic %S: 0.56
Pyritic %S: 1.86

2A,B: Organic %S: 0.33
Pyritic %S: 0.14

Figure III-15. Spectro-chronograms of Argonne coal standards thermally decomposed at 550 C with silica addition (9:1 silica:coal).

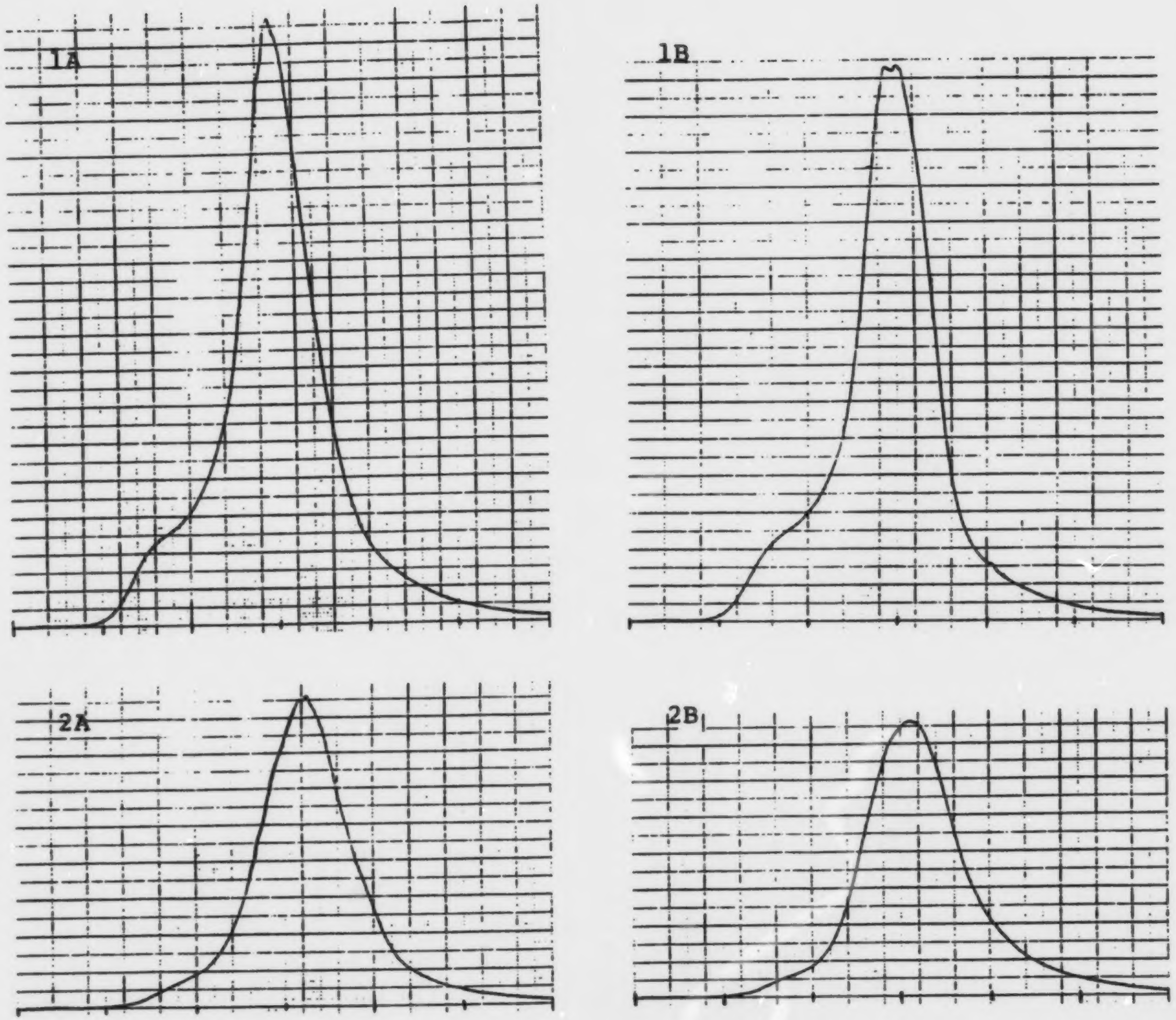


Figure III-16. Spectro-chronograms of Argonne coal standards thermally decomposed at 550 C with silica addition (9:1 silica:coal).

Trace 1A,B: Organic %S: 2.01; Pyritic %S: 2.47.
Trace 2A,B: Organic %S: 0.71; Pyritic %S: 1.32

To investigate reproducibility ten successive analyses were carried out on one type of coal. Results on peak positions and peak height with their standard deviations are summarized in Table III-6. The corresponding spectro-chronograms are shown in Figure III-17. The chronograms were found fairly reproducible in both peak position and peak height with a somewhat larger scatter with respect to actual peak shapes.

The effect of the furnace temperature on the reproducibility and specificity above and below 500°C was also investigated. It was found that the 500 to 550°C region was the best and above 550°C specificity had been gradually lost. The spectro-chronograms for the temperature between 525 and 700°C are displayed in Figures 18 and 19.

Reducing the temperature below the 500°C resulted in a progressively incomplete sulfur combustion with irreproducible chronograms.

From studies on a programmed-temperature oxidation of pyrites and organic sulfur-containing compounds reported in the literature, it was found that maximum SO_2 formation from a non-aromatic sulfide oxidation occurs near 320°C and for aromatic sulfide near 480°C. The maximum SO_2 formation from the pyritic sulfur occurred near 450°C. The temperature-programmed studies of the reduction of sulfur groups in hydrogen atmosphere have also found that the maximum reduction rate of pyrites as a function of temperature falls in between the aliphatic and aromatic sulfur containing organic compounds.

Based on the present study on coals, and coals with added organic sulfur containing compound (see Figure III-20), and the results reported in the literature, it is clear that part of the SO_2 evolution from the organic sulfur cannot be separated from the pyritic sulfur by varying temperature and other means of quantitatively identifying these compounds in coal are necessary.

Table III-6

Chronogramic Parameters for Ten Replicate Samples (100 mg) of Leco Coal Type 8 (4.52% S) Thermally Decomposed at 500°C With Silica Dilution (9:1), (Figure III-17)

Replicate	Sulfur (%)	Peak 1[a] Time (sec)	Peak 1[a] Height (mm)	Peak 2[a] Time (sec)	Peak 2[a] Height (mm)
1	4.27	110	19.0	258	51.5
2	4.12	115	17.0	269	42.0
3	4.24	115	18.0	264	50.5
4	4.14	115	17.5	263	43.5
5	4.18	109	18.5	259	45.5
6	4.22	115	18.5	262	49.0
7	4.18	115	18.5	262	45.0
8	4.22	108	19.5	259	48.5
9	4.37	113	21.0	252	67.0
10	4.30	110	20.0	258	57.5
$\bar{X} \pm S$[b]	4.22 ± 0.07	113 ± 3	18.8 ± 1.1	261 ± 4	50.0 ± 7.1

a. Values represent decomposition time required to yield maximum peak height for each component peak.

b. $\bar{X}$ and S represent the replicate mean and standard deviation, respectively.

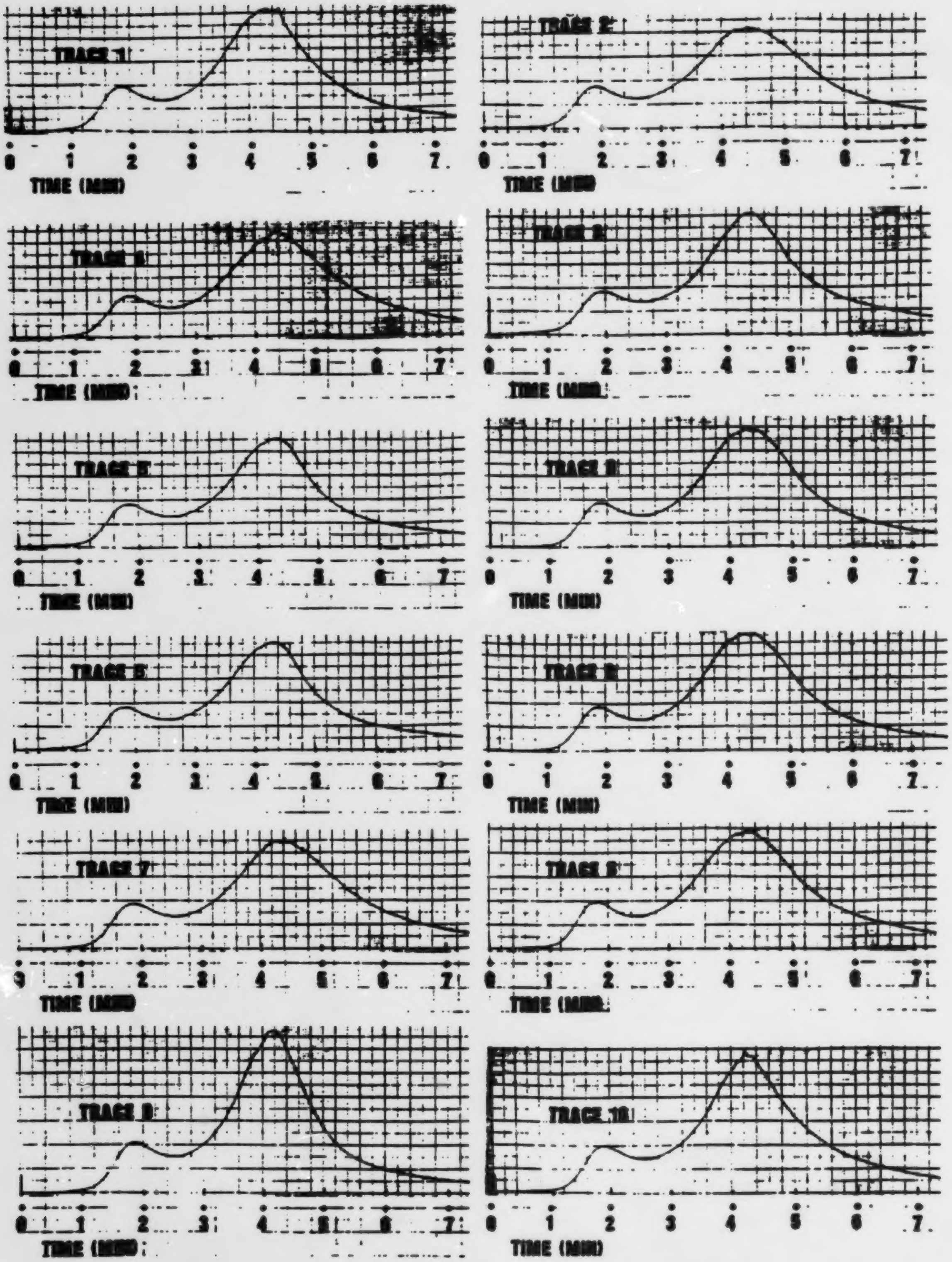


Figure III-17. Replicate spectro-chronograms (n=10) of Leco coal type 8 (4.52% S) thermally decomposed at 500 C with silica addition (9:1 silica:coal).

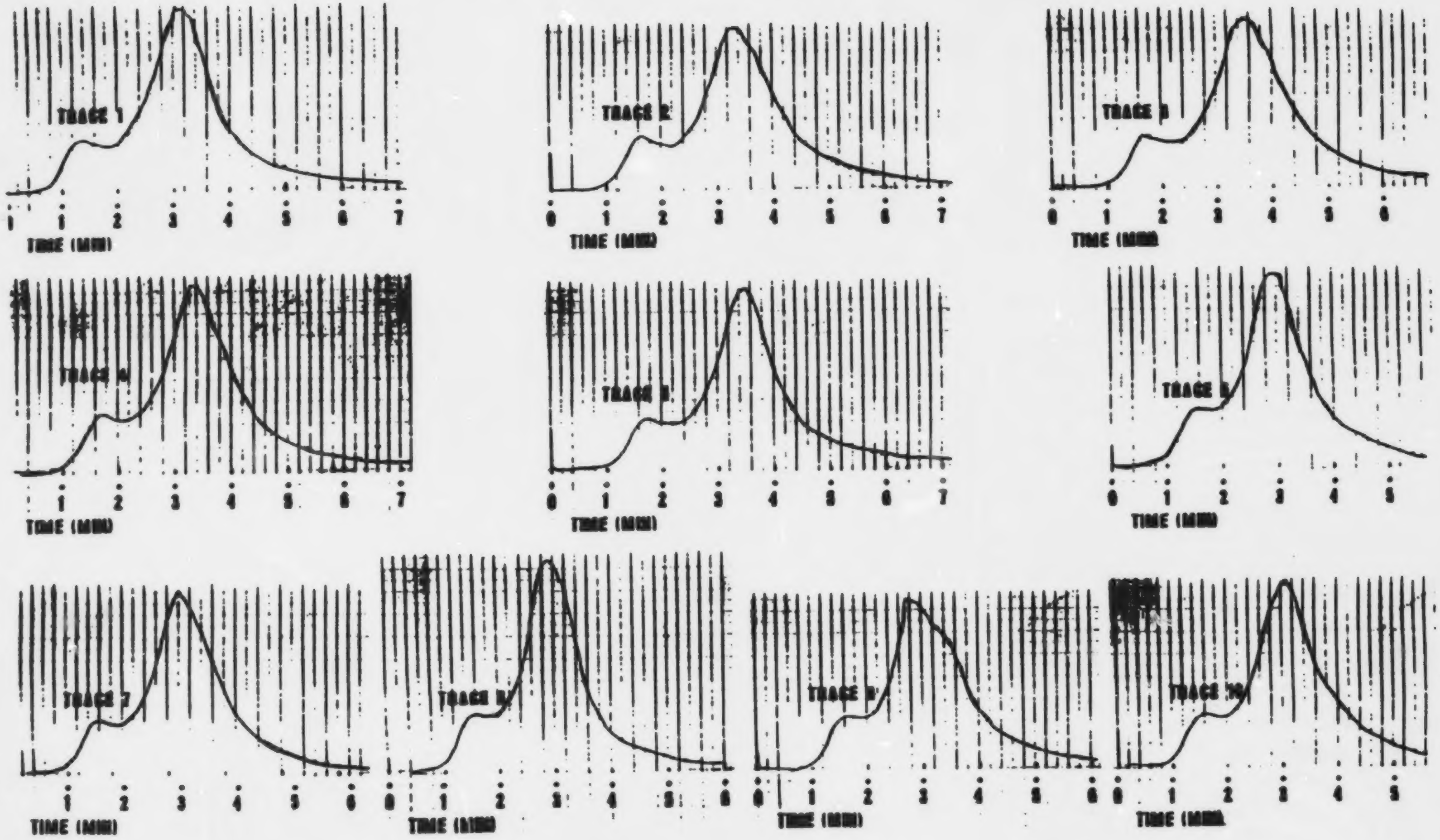


Figure III-18. Replicate chronograms (n=5) of Leco coals type 8 (4.52% S) thermally decomposed at 525 and 550 C with silica addition (9:1 silica:coal).

Traces 1 to 5: 525 C
Traces 6 to 10: 550 C

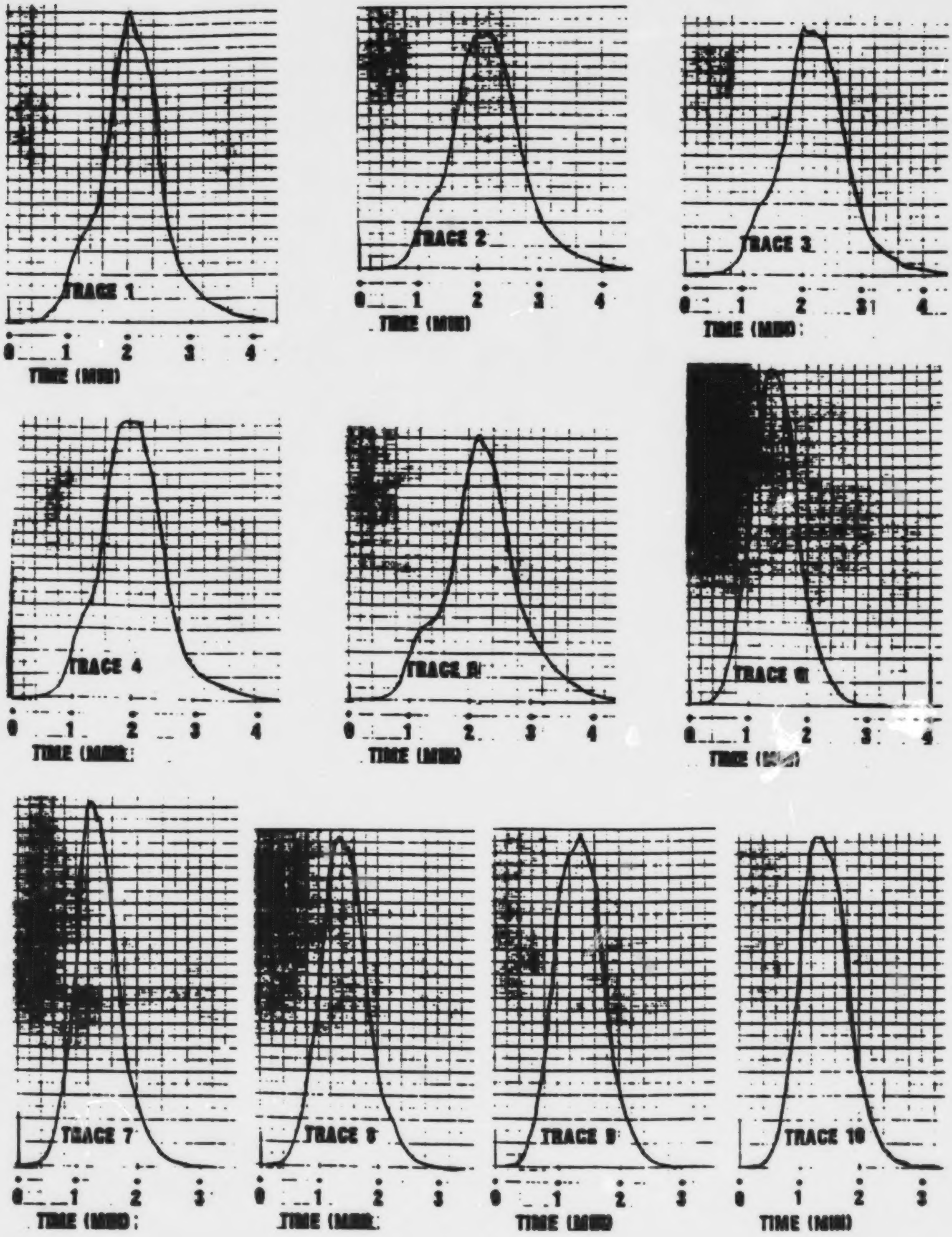


Figure III-19. Replicate spectro-chronograms (n=5) of Leco coal type 8 (4.52% S) thermally decomposed at 600 and 700 C with silica addition (9:1 silica:coal).
Trace 1 to 5: 600 C.
Trace 6 to 10: 700 C.

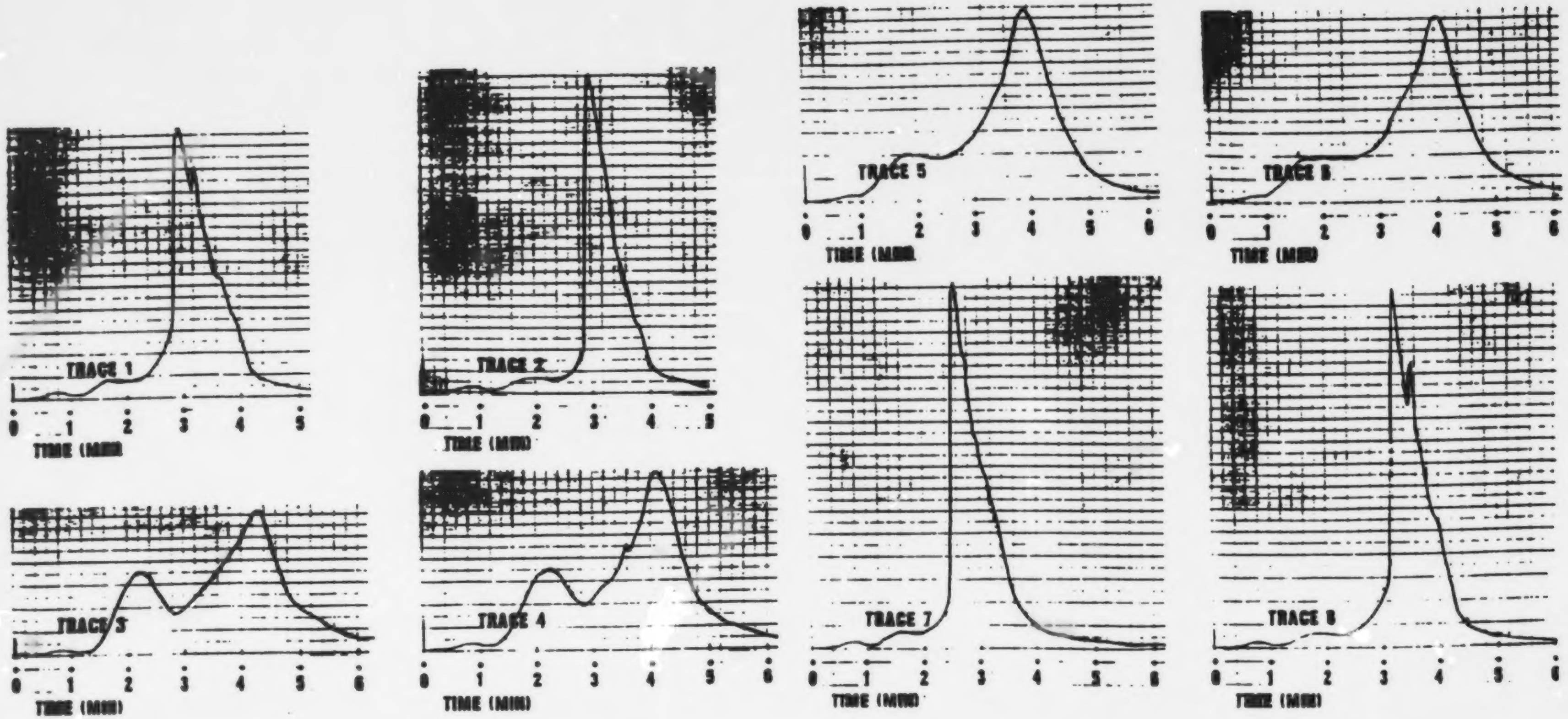


Figure III-20. Chronograms of 100 mg samples of Leco coal type 7 (3.58% S) thermally decomposed at 525 C with silica addition (9:1 silica:coal) and with various organic and pyritic sulfur amendments.

Traces 1 and 2: Coal.
Traces 3 and 4: Coal + 1% Benzene Sulfonic Acid Sulfur.
Traces 5 and 6: Coal + 1% Napthalene Sulfonic Acid Sulfur.
Traces 7 and 8: Coal + 1% Pyrite Sulfur.

Peak deconvolution alone will also not resolve the overlapped SO_2 chronogram formed from aromatic and pyritic sulfides. However, based on calculations performed by K.S. Moon (1987) and paper by Yurovskii (1974) when assumptions are made that the K-value, (ratio between the organic and pyritic SO_2 peak) remains constant then peak deconvolution can result in quantitation of both organic and pyritic sulfur.

IV. PEAK DECONVOLUTION

IV.1 COMPUTER SYSTEMS

The implementation of deconvolution techniques for quantitative analysis of sulfur forms in coal requires the development of real-time data collection and storage of spectro-chronograms on a computer system and the assessment of appropriate processing techniques for separation and quantitation of the sulfur forms.

A desktop microcomputer system was purchased to acquire the raw signal from the Leco SC-132 and store it in files associated with sample runs. A Compaq Deskpro 286 was selected. This computer is equipped with a monochrome monitor, a printer and an analog-to-digital converter card.

In the initial computer work on this project, the Compaq computer was used for acquisition of data only. The data were transferred to a larger computer (SEL 32/77) for the initial deconvolution work. This was done because many of the subroutines needed to perform the deconvolution already existed on the SEL. This provided the fastest and easiest way to do the initial deconvolution work.

Several software programs were written and some hardware modifications were made to the present equipment to accommodate data transfer in real-time from the Leco SC-132 detector to the Compaq Deskpro 286 computer using a Data Translation DT 2811 data acquisition card. Four programs have been written, namely:

1. A C-language program to display data graphically as it is being acquired and to simultaneously store data on a disk file. At the user's option, he/she can redisplay this data after acquisition.

2. A program to transfer data from the Compaq hard disk to 1/2" magnetic tape for data transfer onto the Barringer SEL computer.

3. A program to read the 1/2" tape data into a data base on the SEL computer.

4. A program to read the data base and perform the curve fit.

The software programs required for the real time data acquisition and processing were written and tested. A comparison between the real time spectro-chronograms obtained from the chart recorder analogue output from the Leco SC-132 (Plot A) and the reconstructed spectro-chronogram from the digitized output and computer stored data (Plot B) is displayed in Figure IV-1. The two spectro-chronograms appear identical and, hence, the data acquisition software from the Leco to the Compaq 286, the transfer software from the Compaq 286 to the SEL in-house computer and the computer plotting software were shown to be functioning correctly.

The FORTRAN programs used to perform the peak deconvolution were later transferred from the S.E.L. minicomputer to the Compaq microcomputer.

After the physical transfer of the files, the files still had to be modified somewhat to accommodate slight differences in the FORTRAN compilers resident on the two machines. However, it was found that the programs were quite portable and only a few minor modifications had to be made. The Compaq computer is now a completely self-contained system which can do data acquisition as well as the deconvolution.

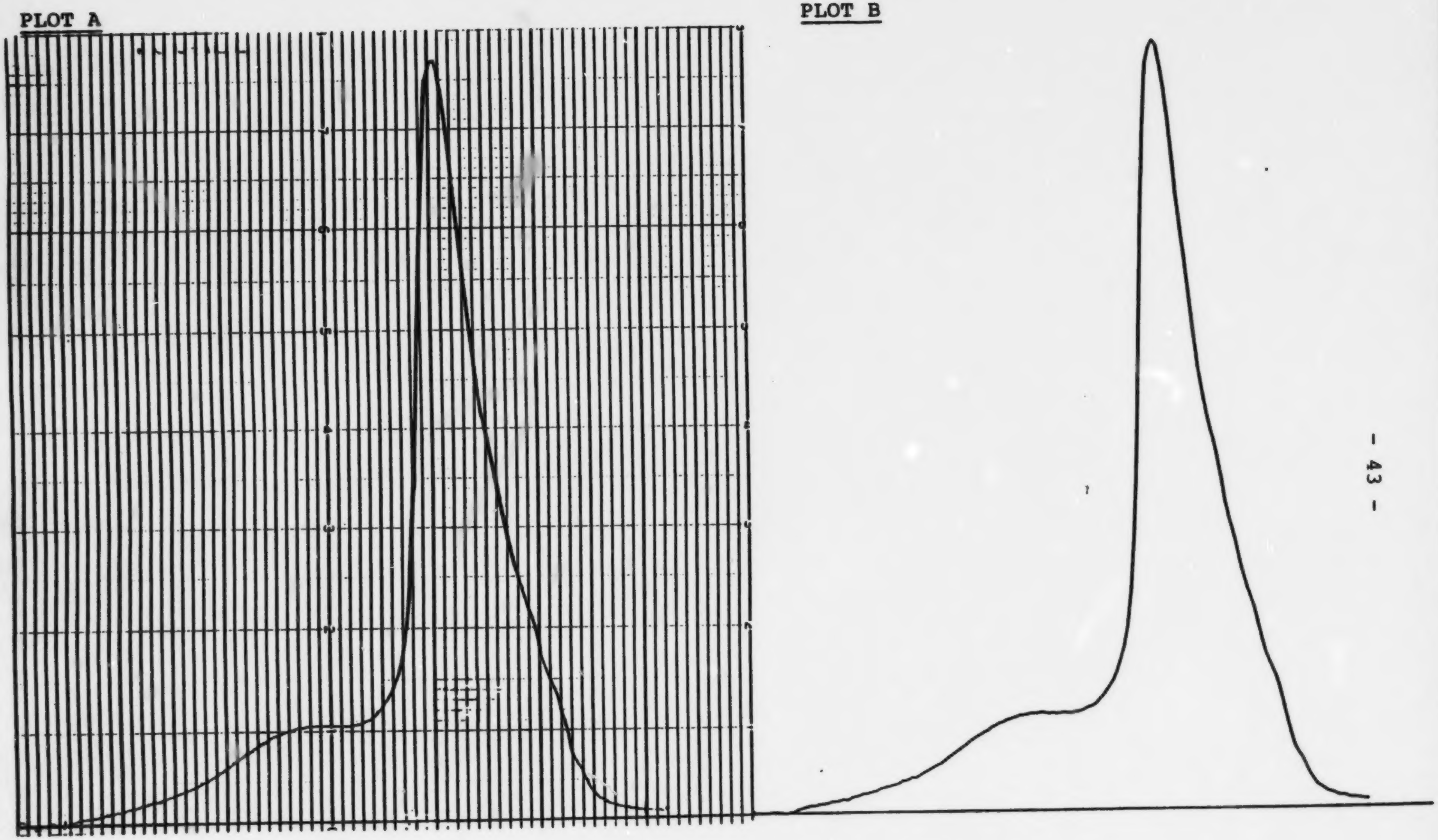


Figure IV-1 Real Time Spectro-Chronogram (Plot A) and Computer Reconstructed Spectro-chronogram From the Microprocessor Acquired Data (Plot B).

A number of utility progams were created on the Compaq to facilitate dealing with the data files. When raw data is acquired into a file, it is embedded among other information. At the beginning of a file, there is a 14-byte header. After that, there are repetitions of a 4-byte time reading and a 2-byte analog-to-digital converter (ADC) reading. A utility program was written to extract the ADC data from a given file and print it on the printer.

A simple program was written to calculate the area under the curve for a given data file. This program takes the first 40 data points, averages them, and considers that average to be the baseline. Then, for every point after that, the accumulated area (with baseline subtracted) is reported. The accumulated area is the total area under the curve from point number 41 to the given point. This is not a very sophisticated peak area determinator and is not meant to replace the deconvolution algorithm. However, it provides a simple means to approximate peak areas. The user must decide over what range of data points to count the areas.

The data acquisition program has the ability to plot the data on the computer monitor as it is being acquired. It also can plot previously acquired data files. For some files, the peaks are small and not much detail can be seen on the monitor. Programs were written to plot the spectrochronograms expanded to 2X and 3X their normal height.

IV.2 COMPUTER MODELS

Marquardt's algorithm does a least squares fit to a non-linear function. It is expected that the peak shapes conform to a Lorentzian (or Cauchy) distribution which can be expressed as:

$$P\ (t, u, \Gamma, A) = \frac{A}{\pi}\ \frac{\Gamma/2}{(t-u)^2 + (\Gamma/2)^2}$$

where t is the independent variable, u is the central value of the peak, and Γ is the full width at half maximum. The amplitude factor, A, allows for peak of different heights.

The spectrochronograms that resulted from the analytical development work generally showed a small first peak and then a larger second peak. The small first peak is due one class of organic sulfur compounds. What appears as the larger peak is actually two severely overlapped peaks, one due to pyrite and the other due to a second class of organic sulfur compounds. Under the experimental conditions used, it is believed that sulfate sulfur either is not liberated or appears in the tail of the pyrite peak. It was considered that the major problem was to resolve the pyrite peak from the organic peaks. Therefore computer models were constructed having two Lorentzian peaks, plus a baseline.

Originally, the program found the best value for mean, width, and amplitude of each of two peaks, as well as the best value for the baseline. This would be an accurate measure of the baseline as long as the baseline does not drift over the course of a run. The true baseline may not drift. However, if there is tailing of the final peak, the signal may not come all the way back to the level of the original baseline before data acquisition is stopped.

Realistically, it is not practical to continue data acquisition for several minutes after the last peak just for the sake of allowing the signal to return the last 1% of the way back to the original baseline position. This means that there may be an apparent drift in the baseline as seen in the acquired data. To account for this, the fitting program was instructed to find the apparent slope of the baseline as an added parameter. Although the slopes that were found by the fitting program were small, by incorporating them into the model, the goodness of the fit was improved in every case in which it was tried.

The model which contains three peaks plus a baseline, plus a contribution from the slope of the baseline can be expressed as:

$$g\ (t, \vec{x}) = f(t, u_1, \Gamma_1, A_1) + f(t, u_2, \Gamma_2, A_2) + f(t, u_3, \Gamma_3, A_3) + B + Mt$$

Given the observed spectrochronogram as input, Marquardt's algorithm finds the best values for the eleven parameters in this equation (mean, width, and amplitude for each of three peaks,, plus baseline, and slope). Work was also done with models having two peaks. In that case, Marquardt's algorithm must optimize eight parameters instead of eleven.

Originally, the deconvolution algorithm was attempting to fit the observed spectrochronograms to a model which has two Lorentzian peaks. A new model was later established in which there are three Lorentzian peaks. Table IV-1 shows a comparison of the goodness of fit (chi squared) between the 2 peak model and the 3 peak model. This is for a series of 14 runs of the same sample,

TABLE IV-1

COMPARISON OF 2 PEAK AND 3 PEAK MODELS

FILE	CHI SQUARED 2 PEAKS	CHI SQUARED 3 PEAKS	
701.032	402.1	130.3	*
701.033	331.4	334.3	*
701.035	241.2	170.8	
701.036	161.3	115.2	
701.037	263.3	162.1	
701.039	153.3	154.6	*
701.040	143.9	45.4	*
701.041	118.3	37.6	
701.042	83.2	83.9	
701.043	139.6	38.0	
701.044	76.7	77.4	
701.045	73.5	23.5	
701.061	64.1	16.1	
701.062	88.4	89.2	

---

* Some parameters reported as negative.

Leco coal standard 501-701. It is apparent that the goodness of fit has improved on going from a model with two peaks to a model that has three peaks. However, as noted in the table, some of the parameters found by the deconvolution algorithm were negative (negative widths and amplitudes for some of the peaks). This, of course, is meaningless. Marquardt's algorithm is simply performing a non-linear regression and has no constraints to prevent such meaningless results. Constraints must be introduced outside of Marquardt's algorithm to prevent this. A further note of caution: The goodness of fit will generally improve as the model gets more complex, regardless of the appropriateness of the model. This is because it is always easier to fit to a model that has more parameters that can be optimized. Therefore, the improvement in chi squared should not be taken as proof that the 3 peak model necessarily is better than the two peak model.

Figures IV-2 to IV-15 show the spectrochronograms for this series of runs.

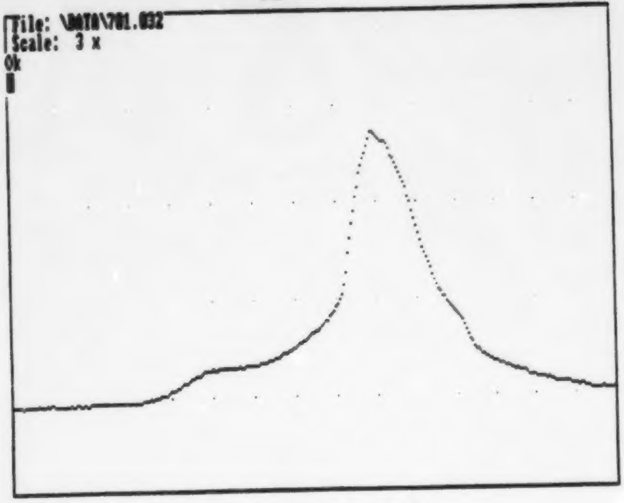


Figure IV-2

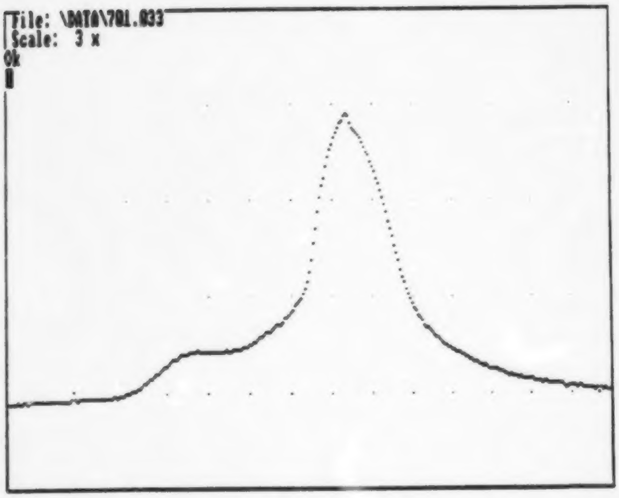


Figure IV-3

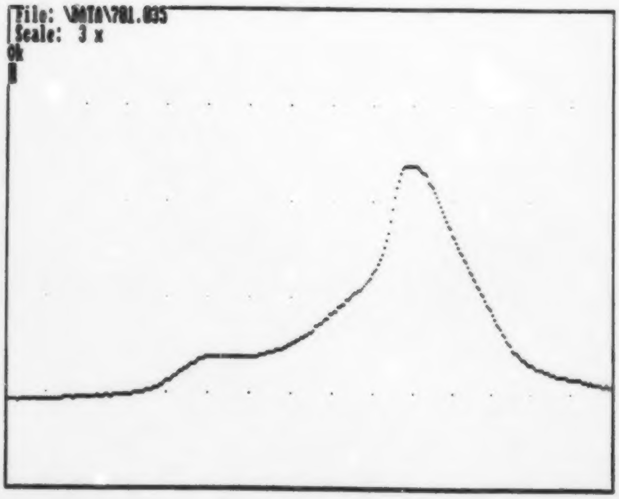


Figure IV-4

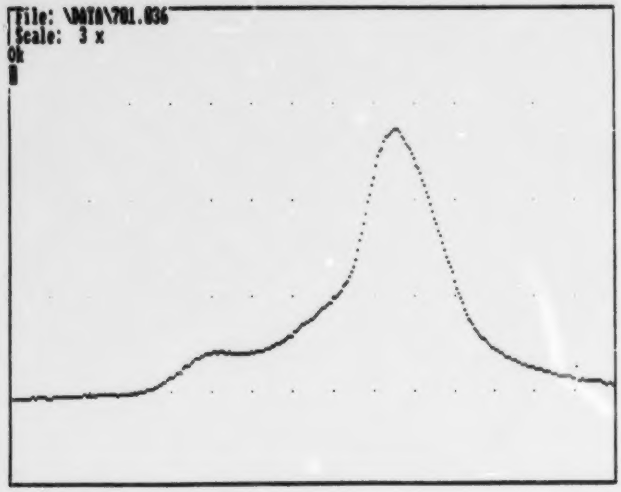


Figure IV-5

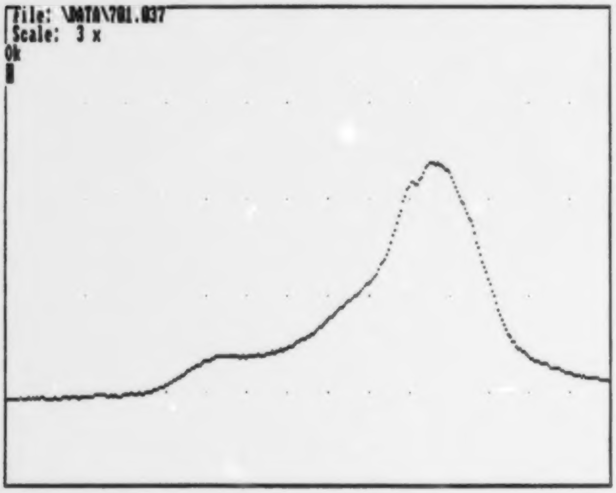


Figure IV-6

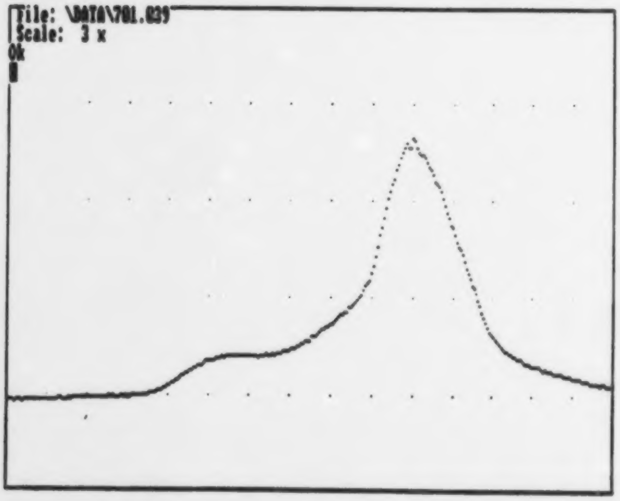


Figure IV-7

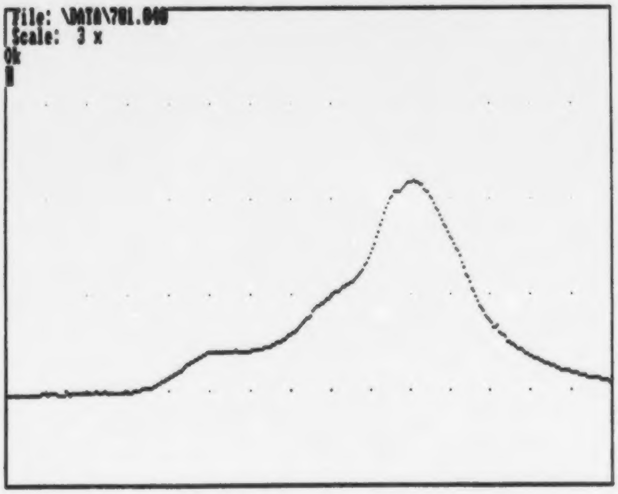


Figure IV-8

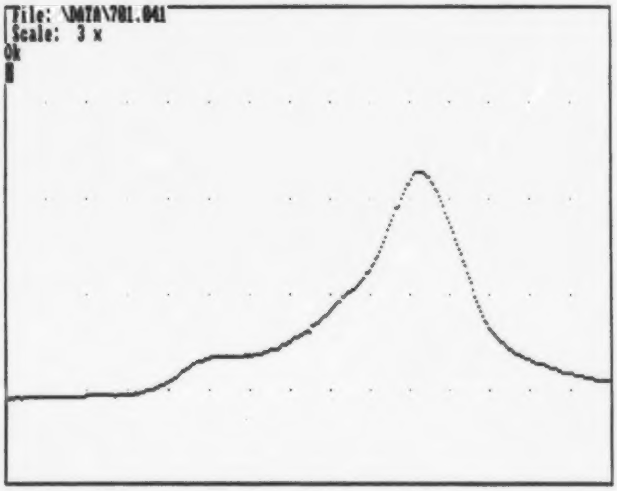


Figure IV-9

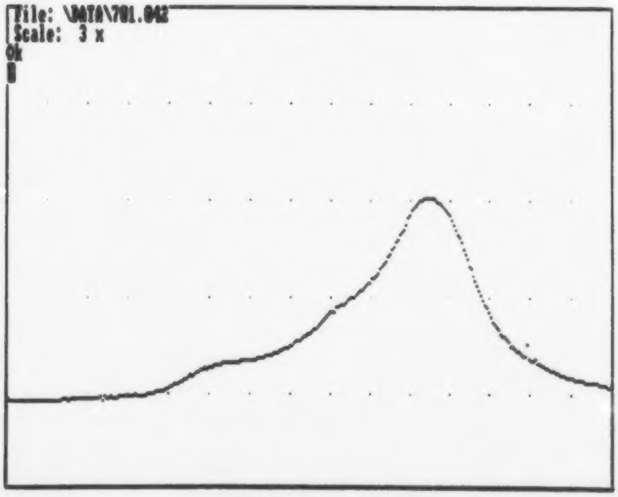


Figure IV-10

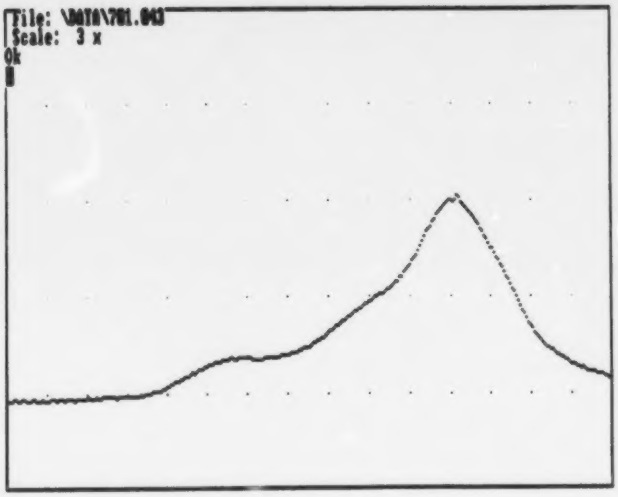


Figure IV-11

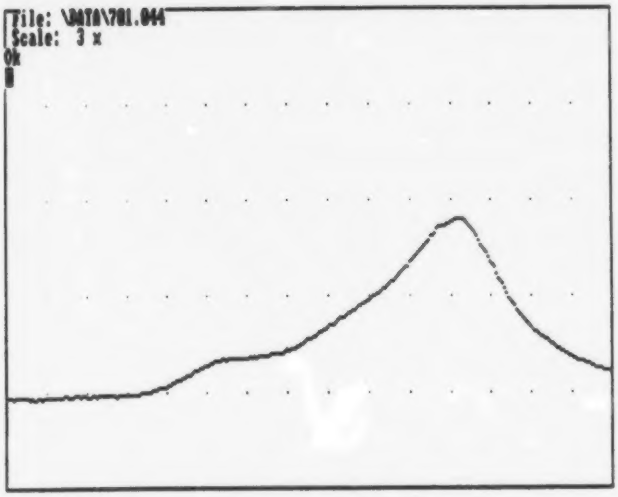


Figure IV-12

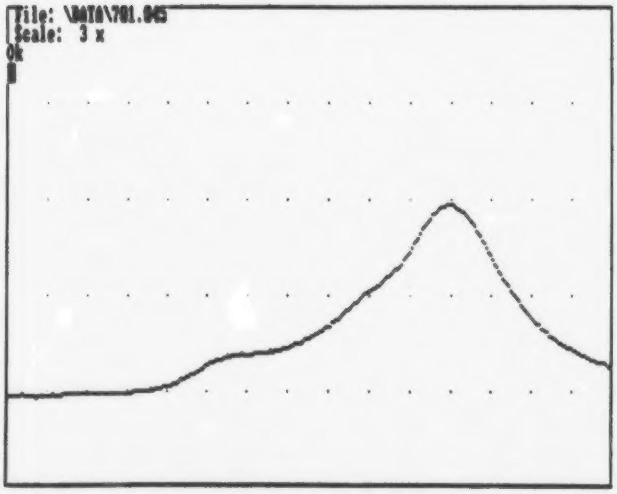


Figure IV-13

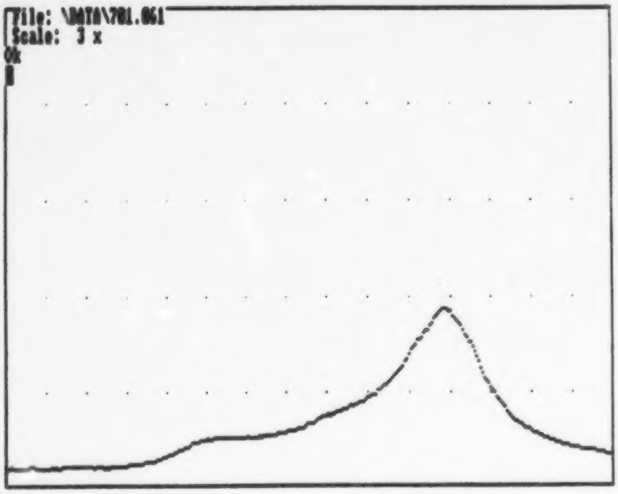


Figure IV-14

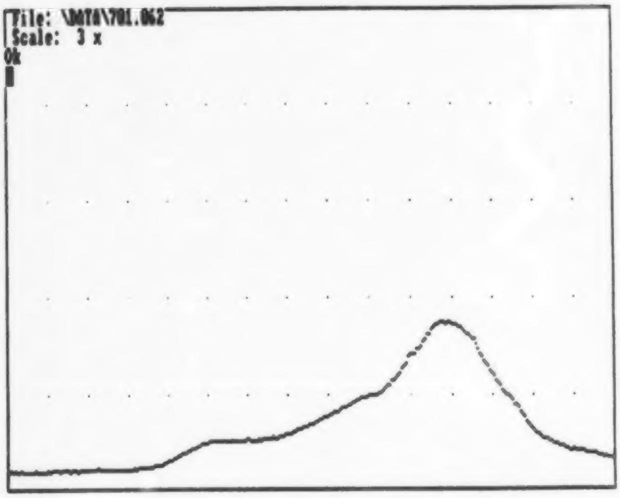


Figure IV-15

IV.3 SYNTHETIC SPECTROCHRONOGRAMS

Marquardt's algorithm was used on the SEL minicomputer and the Compaq microcomputer to do a non-linear least squares fit to the data. In order to check how well the algorithm was working it was useful to generate synthetic spectro-chronograms that were combinations of pure Lorentzian peaks. These were computer-generated on the Compaq computer and put into files in the same format as the observed data. By using these calculated spectro-chronograms, minor bugs in the deconvolution programs were found and corrected. This was necessary before submitting real, observed data to the program.

Table IV-2 lists the parameters used to generate each of the synthetic spectrochronograms. As an example of a bug, an early version of the program took the minimum value in the data file as the baseline and subtracted this from every point. It then used Marquardt's algorithm to find the central value, width, and amplitude for each of two Lorentzian peaks (a total of six parameters to be optimized). This was not very successful. Success of the algorithm could be judged with synthetic data because the true answers are known. Then the program was changed so that the baseline was a seventh parameter that Marquardt's algorithm was to find rather than using a crude initial guess of the baseline value. This made the algorithm much more successful.

Initial estimates for each of the seven parameters must be given to the algorithm. The algorithm's robustness was tested by giving it initial estimates with known errors. For example, when the initial estimates for each of the seven parameters were 20% higher than the true value, the algorithm still found the correct answers. When the initial estimates were each 50% too high, the correct answer was not found. However, with the initial estimates of the two amplitudes 200% too high and the other parameters 20% too high, the correct answers were still found.

TABLE IV-2

PARAMETERS OF SYNTHETIC SPECTROCHRONOGRAMS

FILE	BASELINE	MEAN1	WIDTH1	AMPL1	MEAN2	WIDTH2	AMPL2	LOGICAL RECORD
FAKE1	300	50	20	100,000				44
FAKE2	300	150	50	100,000				45
FAKE3	300	50	50	100,000	150	50	250,000	39
FAKE4	300	100	50	50,000	150	50	250,000	40
FAKE5	300	75	50	75,000	150	50	250,000	41
FAKE6	300	75	50	50,000	150	50	250,000	42
FAKE7	300	75	50	30,000	150	50	250,000	43
FAKE8	300	150	50	100,000	250	50	250,000	
FAKE9	300	750	50	100,000	850	50	250,000	

This indicates that the algorithm is less sensitive to errors in the initial estimates for amplitude than for the other parameters. This is fortunate because amplitude is probably more difficult to estimate accurately than the other parameters.

Because Marquardt's algorithm requires initial estimates for the seven parameters, a subroutine was written to make the initial estimates.

Most of the attempts to resolve the overlapped peaks were done with a Lorentzian model for the peak shape. It has also been attempted with a Gaussian model. In general it has been found that the Lorentzian model is more successful (based on the values of chi squared, the estimate of goodness of fit). This can also be seen visually. Figure IV-16 is the observed spectro-chronogram for an analysis of Leco 501-701. Figure IV-17 shows the best fit calculated for Lorentzian peaks. Figure IV-18 shows the best fit assuming Gaussian peaks.

Figure IV-19 shows another run on the same sample (Leco 501-701). Its Lorentzian best fit is shown in Figure IV-20. In this particular case the algorithm had difficulty in locating the central value of the first peak.

The deconvolution program works in two steps: First, the program makes initial estimates for all the parameters that define the model. Then, Marquardt's algorithm is used to optimize the parameters so as to achieve the best possible agreement between the observed data and the model.

Further synthetic spectrochronograms were generated by specifying two Lorentzian peaks plus a baseline. The separation between the mean values, as well as the widths and amplitudes determined the severity of the overlap between the two peaks. The parameters were selected to approximate observed spectrochronograms in which there is organic and pyritic sulfur. The model used symmetrical

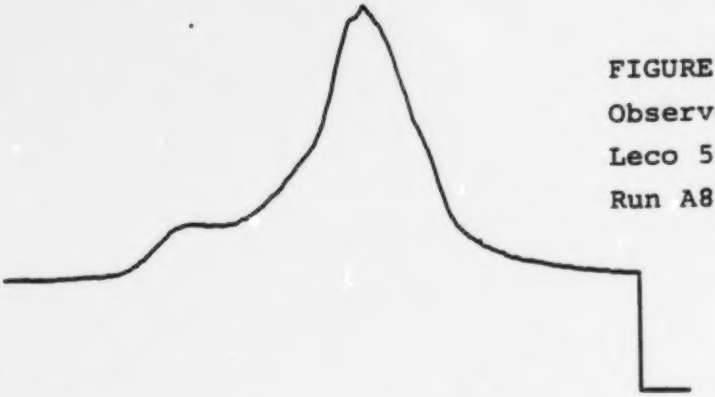


FIGURE IV-16
Observed Spectrochronogram
Leco 501-701
Run A863

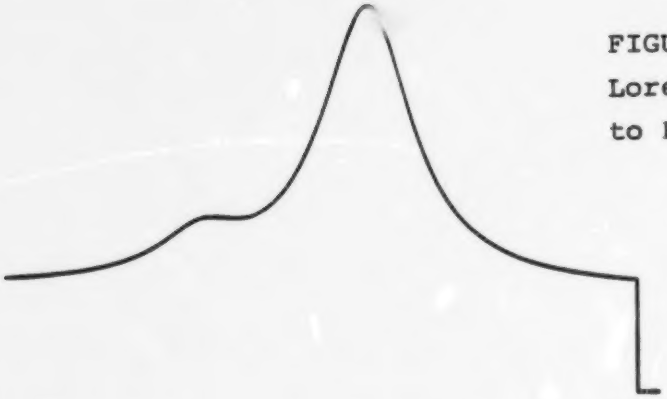


FIGURE IV-17
Lorentzian Best Fit
to Run A863

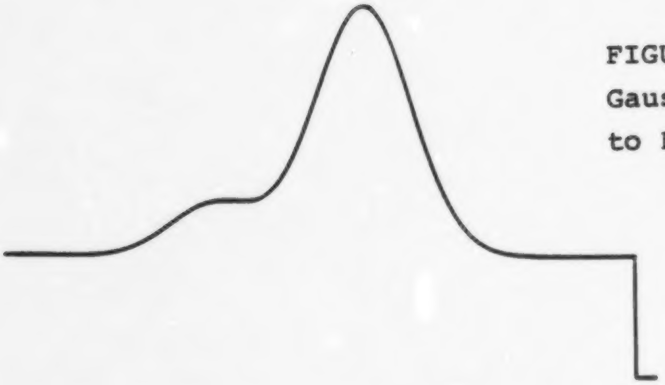


FIGURE IV-18
Gaussian Best Fit
to Run A863

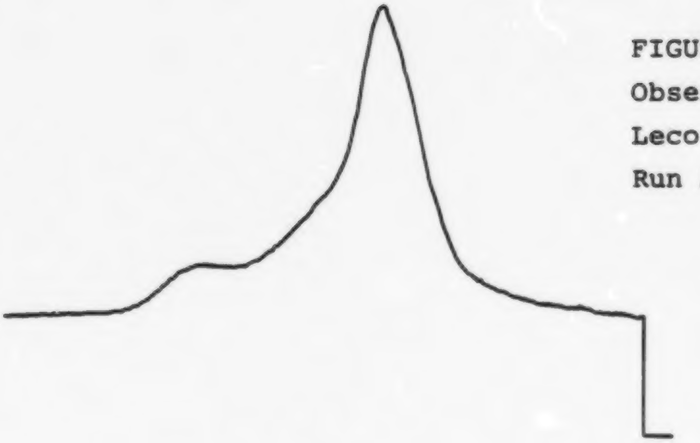


FIGURE IV-19
Observed Spectrochronogram
Leco 501-701
Run A864

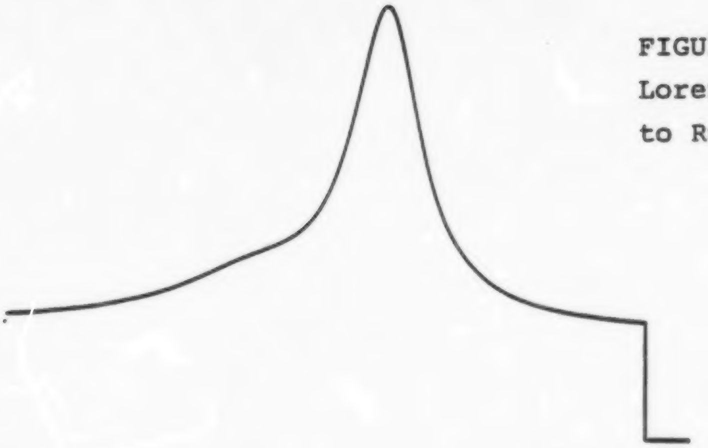


FIGURE IV-20
Lorentzian Best Fit
to Run A864

Lorentzian peaks. The synthetic spectrochronograms are presented in Figures IV-21 to IV-25, along with the true values for all seven parameters.

To test Marquardt's algorithm, initial estimates given as input to the algorithm were set at known deviations from the true values. The results of these tests are shown in Table IV-3. For example, when the initial estimates for the amplitudes of the two peaks were 200% higher than their true values while the initial estimates for the other five parameters were 20% too high, Marquardt's algorithm still found the correct values in four out of five cases. The algorithm is similarly insensitive to very poor estimates of the peak widths (200% too high). However, when the estimates of the peak positions (mean1 and mean2) were in error by 50%, the correct answers were never found. Clearly, it is more important to have a good initial estimate of the peak positions than a good estimate of the peak amplitudes or widths.

The shapes of the observed spectrochronograms suggest that an asymmetrical Lorentzian peak may be a better model for the second peak than a symmetrical Lorentzian peak. Asymmetry is expressed mathematically as having a different width and amplitude on the left side of the peak from those on the right side.

For $x < u$ (left side):

$$P_L = \frac{A_L}{\pi} \cdot \frac{\Gamma_L/2}{(x-u)^2 + (\Gamma_L/2)^2}$$

For $x > u$ (right side):

$$P_L = \frac{A_R}{\pi} \cdot \frac{\Gamma_R/2}{(x-u)^2 + (\Gamma_R/2)^2}$$

TABLE IV-3

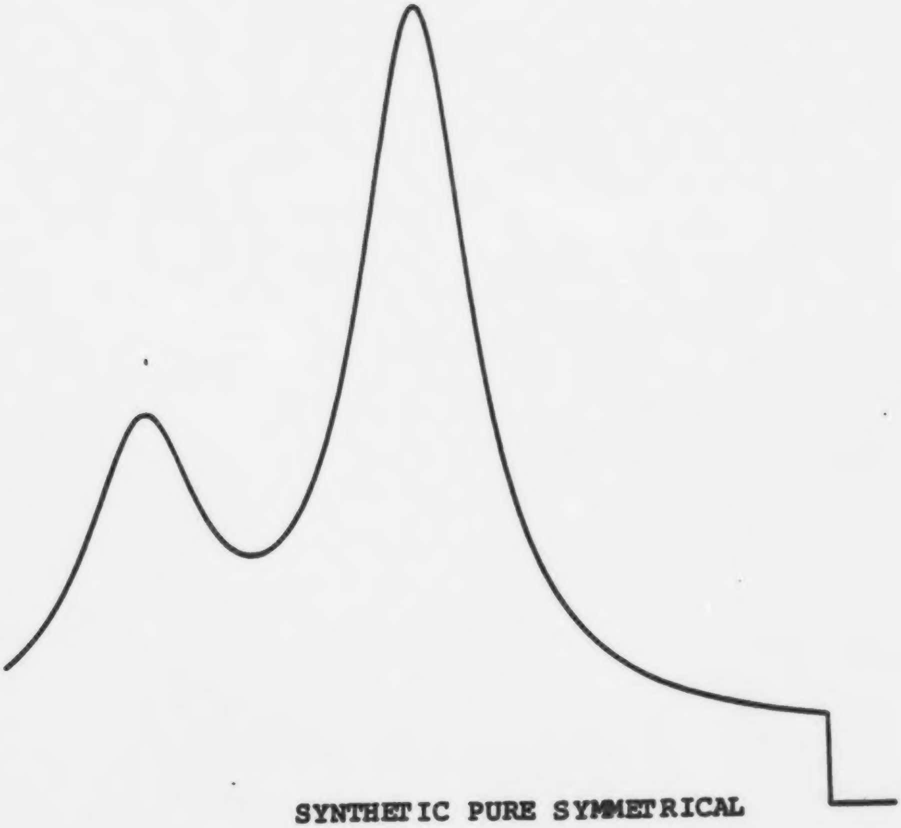
EFFECTIVENESS OF MARQUARDT'S ALGORITHM ON SYNTHETIC DATA WITH PERTURBED INPUTS

Logical Record	All Params. 20% High	All Params. 20% Low	All Params. 50% High	All Params. 50% Low	Amplitudes 200% High Others 20%	Widths 200% High Others 20%	Mean1 50% Low Mean2 50% High Others 20%
39	✓	✓	✓	No	✓	✓	No
40	✓	No	No	No	✓	✓	No
41	✓	No	No	✓	✓	✓	No
42	✓	✓	✓	No	✓	✓	No
43	✓	No	✓	*	No	No	No

✓ - Correct results were found

No - Correct results were not found

* - Absolute value of results correct, sign wrong

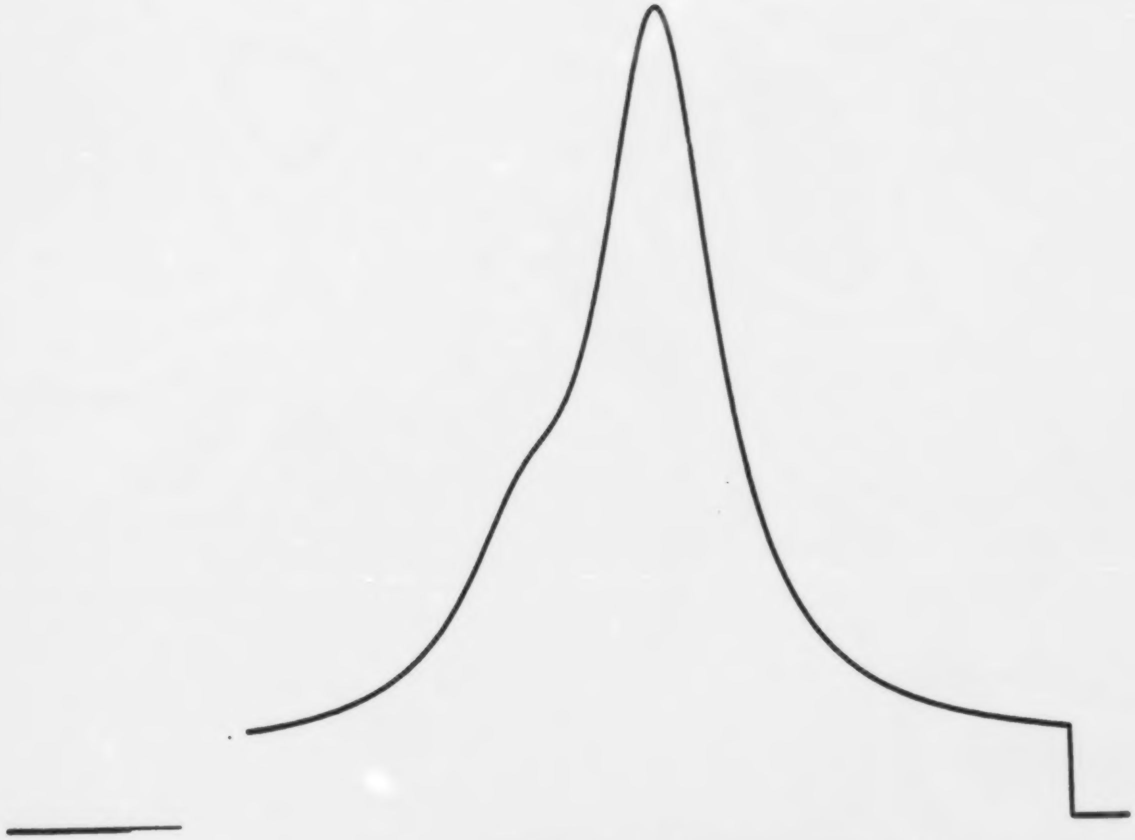


SYNTHETIC PURE SYMMETRICAL

LORENTZIANS

Logical Record:	39
True Values:	
Mean 1:	50
Width 1:	50
Amplitude 1:	100,000
Mean 2:	150
Width 2:	50
Amplitude 2:	250,000
Baseline	300

FIGURE IV-21



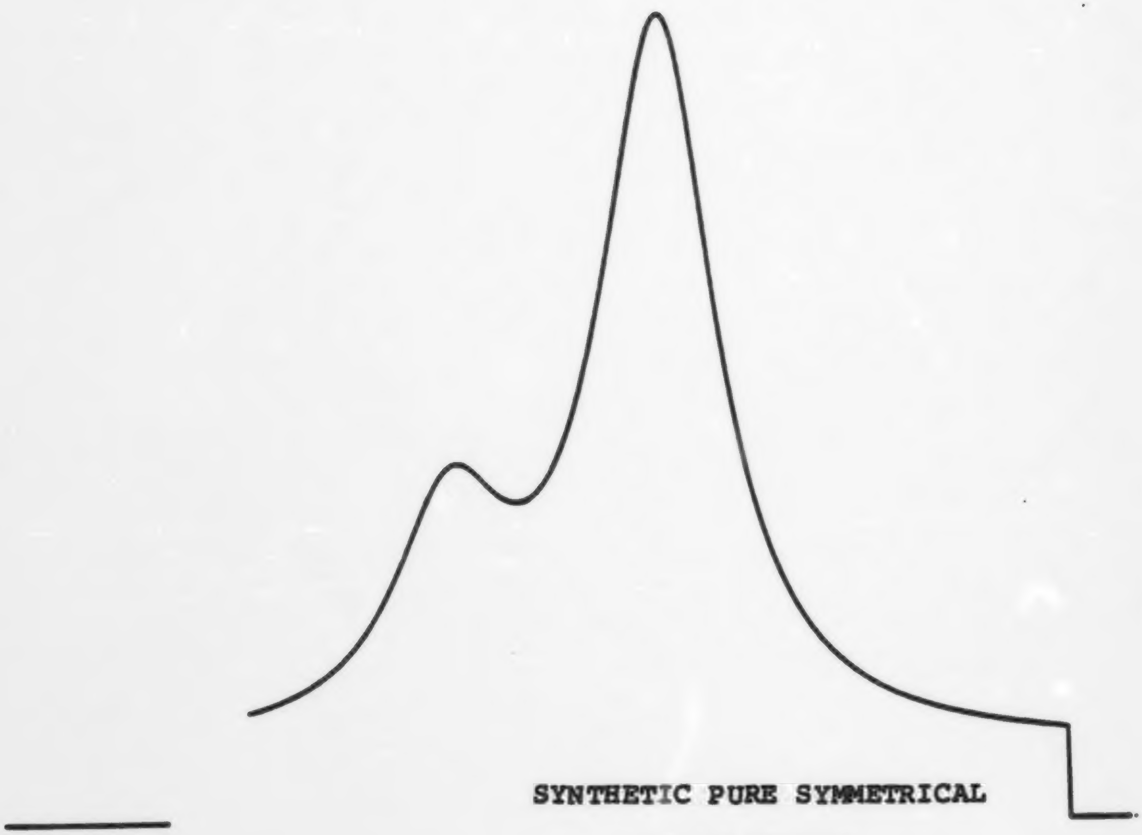
SYNTHETIC PURE SYMMETRICAL

LORENTZIANS

Logical Record: 40
True Values:

Mean 1: 100
Width 1: 50
Amplitude 1: 50,000
Mean 2: 150
Width 2: 50
Amplitude 2: 250,000
Baseline 300

FIGURE IV-22

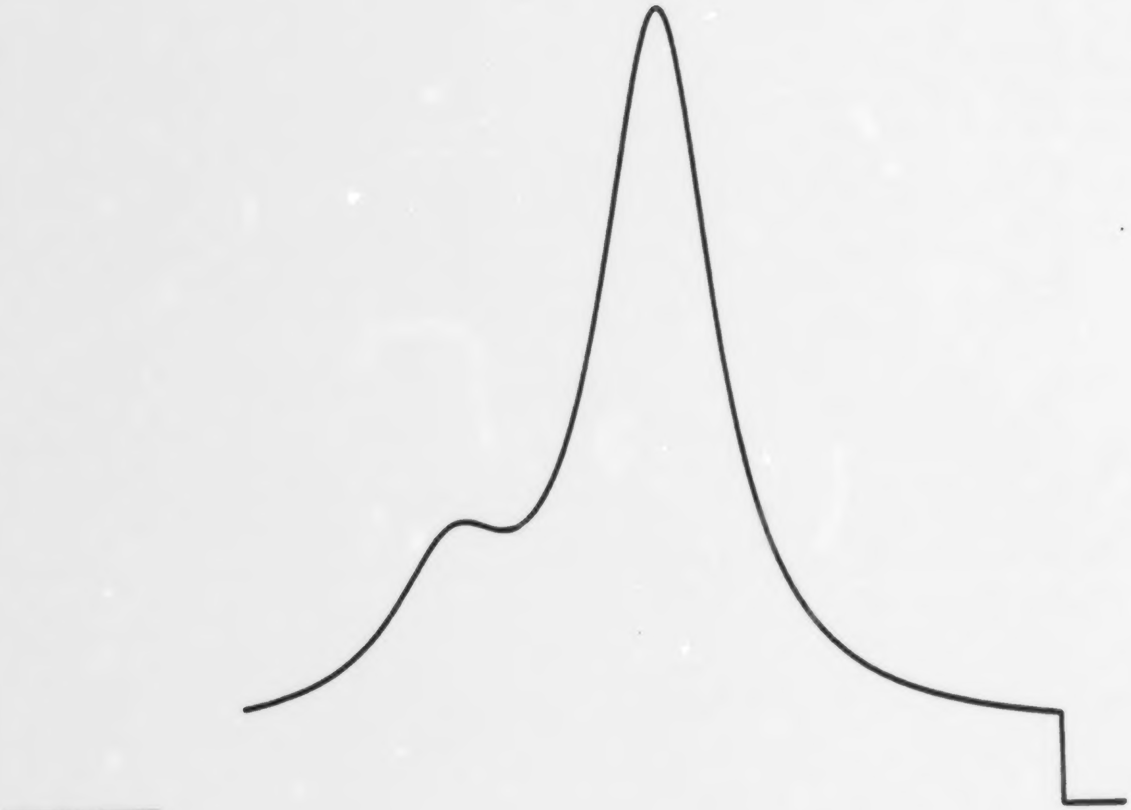


SYNTHETIC PURE SYMMETRICAL

LORENTZIANS

Logical Record:	41
True Values:	
Mean 1:	75
Width 1:	50
Amplitude 1:	75,000
Mean 2:	150
Width 2:	50
Amplitude 2:	250,000
Baseline	300

FIGURE IV-23

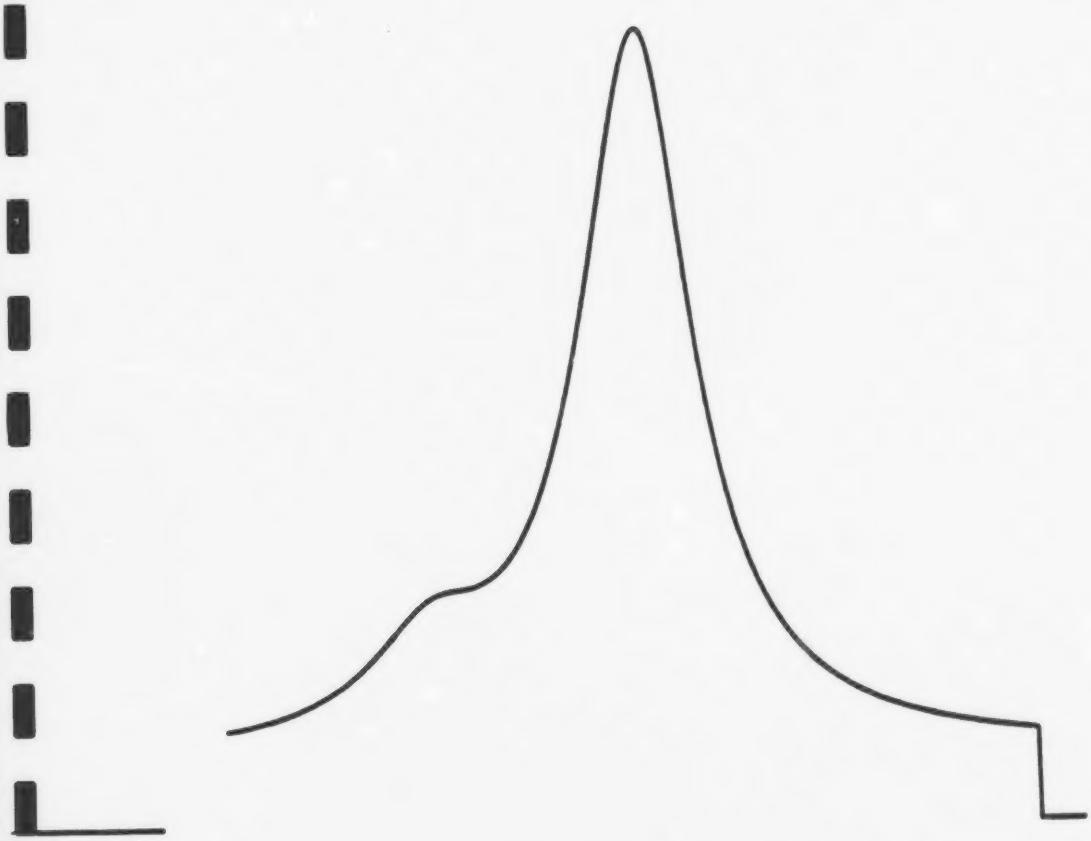


SYNTHETIC PURE SYMMETRICAL

LORENTZIANS

Logical Record:	42
True Values:	
Mean 1:	75
Width 1:	50
Amplitude 1:	50,000
Mean 2:	150
Width 2:	50
Amplitude 2:	250,000
Baseline	300

FIGURE IV-24



SYNTHETIC PURE SYMMETRICAL

LORENTZIANS

Logical Record:	**43**
True Values:	
Mean 1:	**75**
Width 1:	**50**
Amplitude 1:	**30,000**
Mean 2:	**150**
Width 2:	**50**
Amplitude 2:	**250,000**
Baseline	**300**

FIGURE IV-25

where L and R represent left and right, respectively. The requirement that the two parts of the curve meet in the middle leads to the relation:

$$A_L = A_R \cdot \frac{\Gamma_L}{\Gamma_R}$$

Synthetic spectrochronograms were created with a single asymmetrical Lorentzian peak. File FAKE15.001 had: Mean1=150, Width_L = 100, AMPL_L = 400,000, Width_R = 50, and AMPL_R = 200,000. FAKE16.001 was the same as FAKE15.0001 except that width_R = 25 and AMPL_R = 100,000.

It is readily apparent from plots of observed and fitted data that Marquardt's algorithm had difficulty in locating peak1, the small first peak in the spectrochronogram. Even when the initial estimate of mean1 was very close to the correct value, the final value for mean1 found by Marquardt's algorithm was incorrect and usually it had drifted in the direction of peak2. This is an indication that the model being used was still not a very good description of the real data. The program was modified so as to keep mean1 fixed and not have Marquardt's algorithm optimize it; it would optimize only the other parameters. This improved the fit.

IV.4 PYRITE STANDARD ADDITION STUDY

In situations where there is a strong interference preventing an accurate direct measurement of an instrument's response to an analyte the standard addition method may be used. Known amounts of the analyte are added to the sample matrix so that a plot can be constructed relating instrument response to amount of analyte added. This plot is extrapolated back to zero analyte added to infer the instrument's response for the analyte in the original, unmodified sample. An implicit assumption in this correction for matrix effects is that the response varies linearly with the amount of analyte or at least varies in a way that can be well characterized. This matrix effect situation applies in the case of trying to find the area of the pyrite peak while it is severely overlapped with the second organic peak. A standard addition analysis was done by adding known amounts of pyrite to known amounts of a coal standard. The mixtures were made by adding 10, 20 and 30 mg of FeS_2 to 990, 980, and 970 mg of the coal standard respectively.

A standard addition analysis was performed on Leco coal standard 501-701 which is reported by Leco to have 0.71% organic sulfur, 0.38% pyritic sulfur, and 0.02% sulfatic sulfur. Two runs were done on the sample with no added pyrite. Additions were made by adding pyrite to the coal standard. Before analysis, the sample of 100 mg was mixed with approximately 1 g of silica. The purity of the pyrite was determined by analyzing the pyrite by itself with silica at 1370°C.

Fifty-two milligrams of the pyrite was mixed with 950 mg silica. Three 100 mg aliquots of that mixture were run on the Leco yielding results with a mean of 2.45 ± 0.08%S. Pure pyrite would be 2.77%S. Therefore, the purity of the pyrite is 2.45/2.77 = 88% pure.

Marquardt's algorithm was used to find the area of the second peak (the peak that is due to organic and pyritic components). The results of this standard addition are presented in Table IV-3A and Figure IV-26. The acquired spectrochronograms appear in Figures IV-27 to IV-34.

IV.5 APPLICATION OF CONSTRAINTS TO COMPUTER MODELS

When Marquardt's algorithm was used for the deconvolution of the overlapped peaks, occasionally meaningless results were obtained for some parameters. e.g. negative values of peak position. This is because Marquardt's algorithm tries only to maximize the goodness of fit, i.e, minimize chi-squared, without regard to how reasonable the results may be. To prevent obviously wrong parameters constraints were applied to the values that the parameters can assume.

The constraints were applied between consecutive iterations of the Marquardt algorithm program. After each iteration the main program checked certain conditions of the parameters and made adjustments in them as necessary. The constraints were designed to eliminate meaningless results, e.g. negative values of peak position, width, or amplitude and also improbable results, e.g., unreasonably large values of peak width or amplitude. The application of the various constraints was an attempt to improve the fit produced by the deconvolution algorithm.

Each set of constraints was applied to the same raw data files (samples of Leco standards 501-701(and their effect on the goodness of fit (chi squared) is shown in Table IV-4. The model assumes 3 Lorentzian peaks where peak1 is the small, first organic peak, peak2 is the large peak due to pyrite, and peak3 is the organic peak which is severely overlapped with, and hidden by the pyrite peak. The definitions of the constraints are:

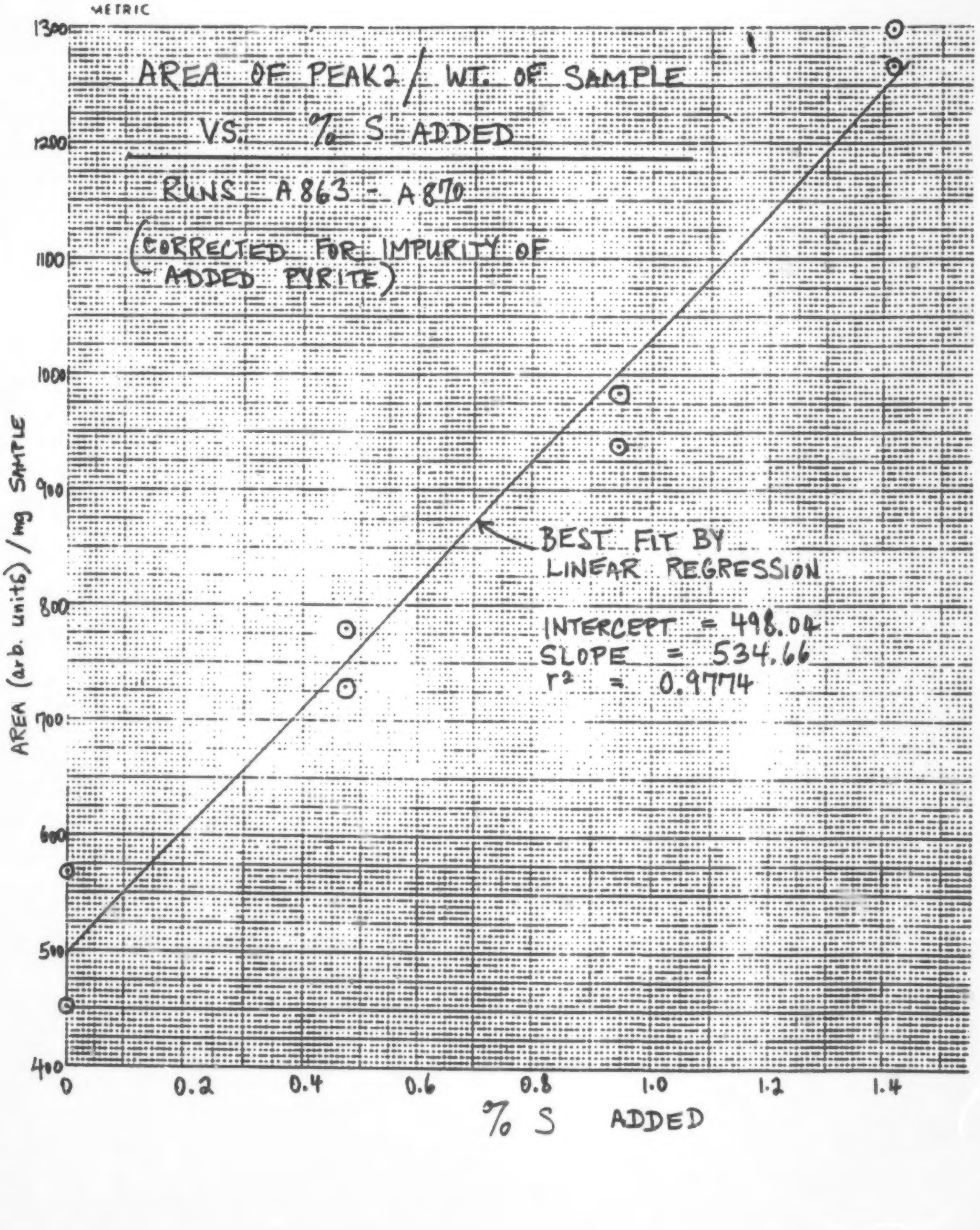
Constraints A: If mean3 > mean2 or mean3 < mean1, then reset mean3 to its initial value. If width3 > 300, then reset width3 to its initial value. Convert all negative parameters to 0.

TABLE IV-3A

Standard Addition Analysis of Leco Coal Standard 501-701

File	Sample Weight (mg)	Area of Peak 2	Area Weight	% S Added
701.020	101	57282	567.2	0
701.021	100	45287	451.8	0
701.022	101	73665	729.1	0.472
701.023	104	81064	779.5	0.472
701.024	102	102224	984.0	0.945
701.025	103	96638	938.2	0.945
701.026	100	129569	1297.7	1.417
701.027	100	127910	1267.3	1.417

FIGURE IV-26


METRIC
AREA OF PEAK2 / WT. OF SAMPLE
VS. % S ADDED
RUNS A863 - A870
(CORRECTED FOR IMPURITY OF ADDED PYRITE)
BEST FIT BY LINEAR REGRESSION
INTERCEPT = 498.04
SLOPE = 534.66
r² = 0.9774
AREA (arb. units) / mg SAMPLE
1300
1200
1100
1000
900
800
700
600
500
400
0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
% S ADDED

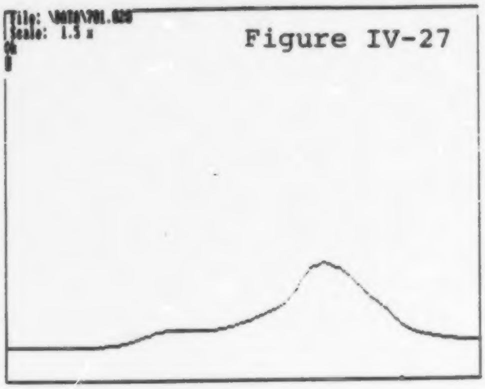


Figure IV-27

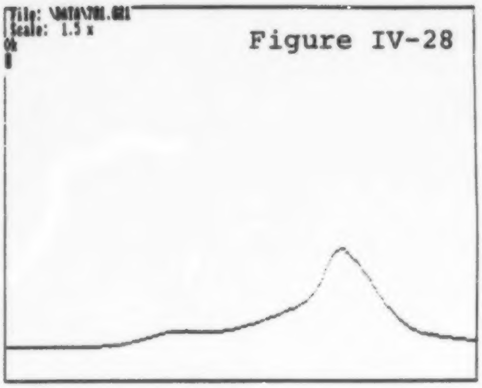


Figure IV-28

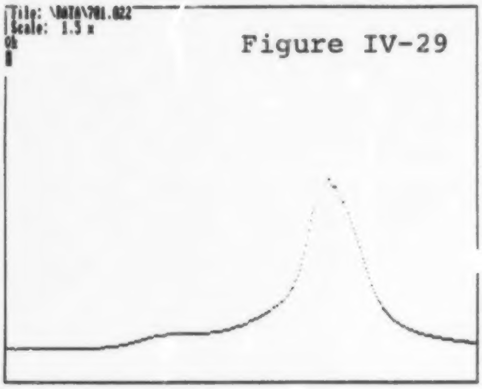


Figure IV-29

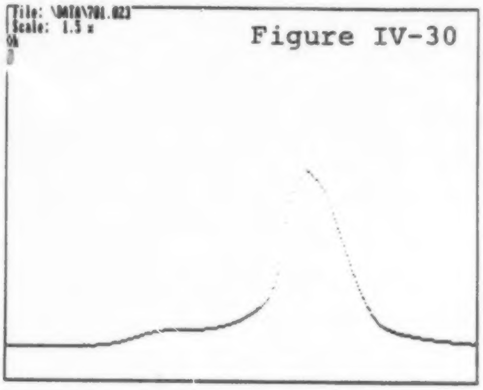


Figure IV-30

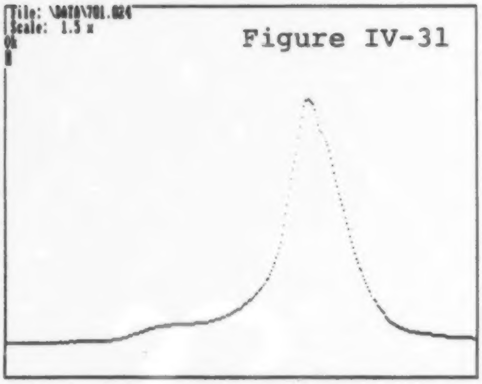


Figure IV-31

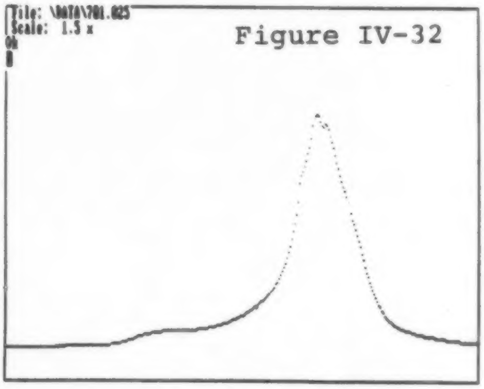


Figure IV-32

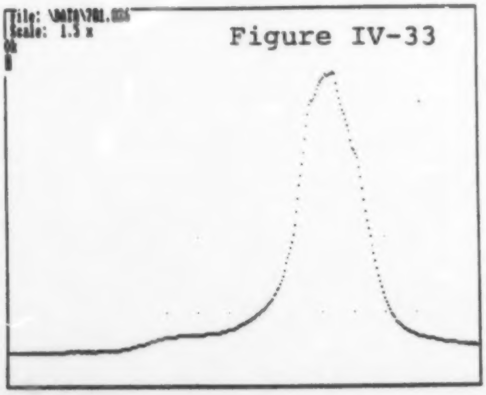


Figure IV-33

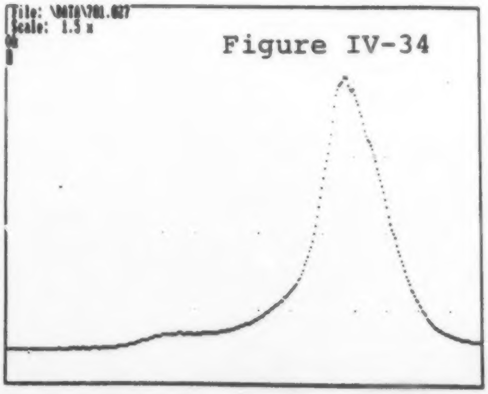


Figure IV-34

Constraint B: Same as constraint A except convert all negative parameters to their absolute values.

Constraints C: Convert all negative parameters to their absolute values. If mean1 $>$ 300, reset mean1 to its initial value. If amplitude1 $>$ 200,000, reset to initial value. If width1 300, reset to initial value. If mean3 $>$ mean2 or mean3 $<$ mean1, then reset mean3 to its initial value. If amplitude3 $>$ 200,000, reset to its initial value. If width3 $>$ 300, reset to its initial value.

Using the 3 peak model, the fitting program must optimize 11 parameters (3 for each peak plus slope and offset of baseline). To execute the deconvolution for the 14 files shown in Table I using constraints C took 34 minutes and 51 seconds. To speed up the calculations, the deconvolutions were done again but with every second data point ignored. The results of this are shown in the rightmost column in Table IV-4. This took only 18 minutes and 17 seconds, effecting a 48% improvement in speed. A comparison of the chi squared in the last 2 columns shows a slight deterioration in 3 cases, and a large deterioration in 1 case. The conclusion was drawn that if computation time becomes an important issue, it may be possible to achieve useable results with half of the current sampling frequency. However, with the current frequency of 1 sample per second, the calculations took an average of 2 1/2 minutes for each run ad this was not considered an excessive time . It is not worth it to take the chance on getting a worse fit to reduce the 2 1/2 minute calculation time.

It was later noted that not all the parameters must be positive. Specifically, the slope of the baseline could logically be positive or negative. Therefore the constraint forcing the slope to be positive was removed.

TABLE IV-4

CHI SQUARED FOR VARIOUS CONSTRAINTS

FILE	CONSTRAINTS A	CONSTRAINTS B	CONSTRAINTS C	CONSTRAINTS C 1/2 SAMPLING FREQUENCY
701.032	396.0	396.3	396.3	600.4
701.033	321.9	341.7	341.7	333.3
701.035	170.7	170.7	170.7	177.1
701.036	115.0	115.0	115.0	116.7
701.037	162.1	162.1	162.1	166.3
701.039	127.3	127.3	127.3	133.7
701.040	172.4	45.5	45.5	47.2
701.041	37.8	37.8	37.8	38.1
701.042	25.2	25.2	25.2	22.6
701.043	38.0	38.0	38.0	38.5
701.044	141.6	17.2	17.2	18.8
701.045	Math Error	23.5	23.5a	25.9
701.061	16.1	Math Error	16.1	16.8
701.062	31.3	31.3	31.3	32.7

Math Error = run time error resulting from math coprocessor exceptions

Measurements of goodness of fit are useful for comparisons among various versions of a fitting algorithm. However, we want to use the fitted curve to determine the individual species of sulfurr in a coal sample and the success of an algorithm in this regard must be the ultimate criterion. Table IV-5 shows the areas found for each of the peaks using Constraints C in this standard (Leco 501-701), organic sulfur is 0.71%, pyritic sulfur is 0.38%, and sulfate sulfur is 0.02%. If the organic sulfur gives rise to peaks 1 and 3 and pyritic gives rise to peak2, then the ratio of (area1 + area3) to area2 should be equal to 0.71/0.38 = 1.87 (neglecting the contribution of sulfate). The last column in Table IV-5 gives the actual values found for this ratio. The fact that these values are far from the expected value and are not reproducible among themselves indicates that this version of the algorithm applied to these data does not have the desired results.

IV.6 THE K-FACTOR METHOD

The attempts to resolve the spectrochronograms into three peaks produced results that were inconsistent. The overlap of the SO_2 peak from aromatic compounds with the SO_2 peak from pyrite is too severe. An alternative approach is described in publications by Moon (1987) and Yurovskii (1974). There is empirical evidence that for a given coal bed, the ratio of sulfur from organics in the first peak to sulfur from pyrite in the second peak is a constant; $k = org_1/pyr$. The seven spectrochronograms shown in Figure IV-35 were acquired on Leco Standard 501-251 which has the following ultimate analysis of sulfur forms:

organic	0.73%
pyritic	1.42%
sulfate	0.10%
Total Sulfur	2.25%

TABLE IV-5

PEAK AREAS FOUND USING CONSTRAINTS C

FILE	AREA1	AREA2	AREA3	$\frac{\text{AREA1+AREA3}}{\text{AREA2}}$
701.032	1551	48618	1150	0.0556
701.033	4164	52302	11	0.0800
701.035	5008	51534	4168	0.1781
701.036	4861	52551	2251	0.1353
701.037	4367	47439	11990	0.3448
701.039	5194	50412	1893	0.1406
701.040	4529	50696	5396	0.1958
701.041	3128	40443	16003	0.4730
701.042	2976	39540	14715	0.4474
701.043	5236	45548	9891	0.3321
701.044	2974	39587	16228	0.4851
701.045	3510	50913	7945	0.2250
701.061	2707	26308	13234	0.6059
701.062	3097	32305	9547	0.3914

The results of the deconvolution program are presented in Table IV-6. The ultimate analysis predicts that 0.73/2.25 x 100% = 32.4% of the total area should be due to organic sulfur. For these 7 runs, the mean ratio of area1 to the total area is 16.13%. This means that 0.1613 x 2.25% = 0.363% S is due to the first peak (org1). Therefore, 0.73% - 0.363% = 0.37% S would be due to the second organic peak (org2).

Ultimate analyses were performed by both CANMET and Leco on five Leco standards and the results of these are presented in Table IV-7. Given the ultimate analysis for a coal sample and the area of peak1 and the total area, it is possible to calculate the k-factor, i.e., the ratio of the area for organic in peak1 to the area for pyrite in peak2. For example, if the ultimate analysis states that pyritic sulfur is 0.33% and total sulfur is 0.97%, then the area due to pyritic S will be 0.33/0.97 or 34% of the total area. The constant k is then calculated as the ratio of area1 to the area due to pyrite. In this way, k was calculated for a number of runs on the five different Leco standards using the ultimate analyses by both Leco and CANMET. Table IV-8 shows the k factors that were found.

The program DECON, which does the deconvolution, reports area1, area2, and the total area. It then asks the operator for the value of k and calculates the % organic S and the % pyritic S based on the k value. It asks for another value of k and calculates results based on that as well. The results given by the DECON program are presented in Table IV-9.

FIGURE IV-35

SPECTROCHRONOGRAMS OF LECO 501-251

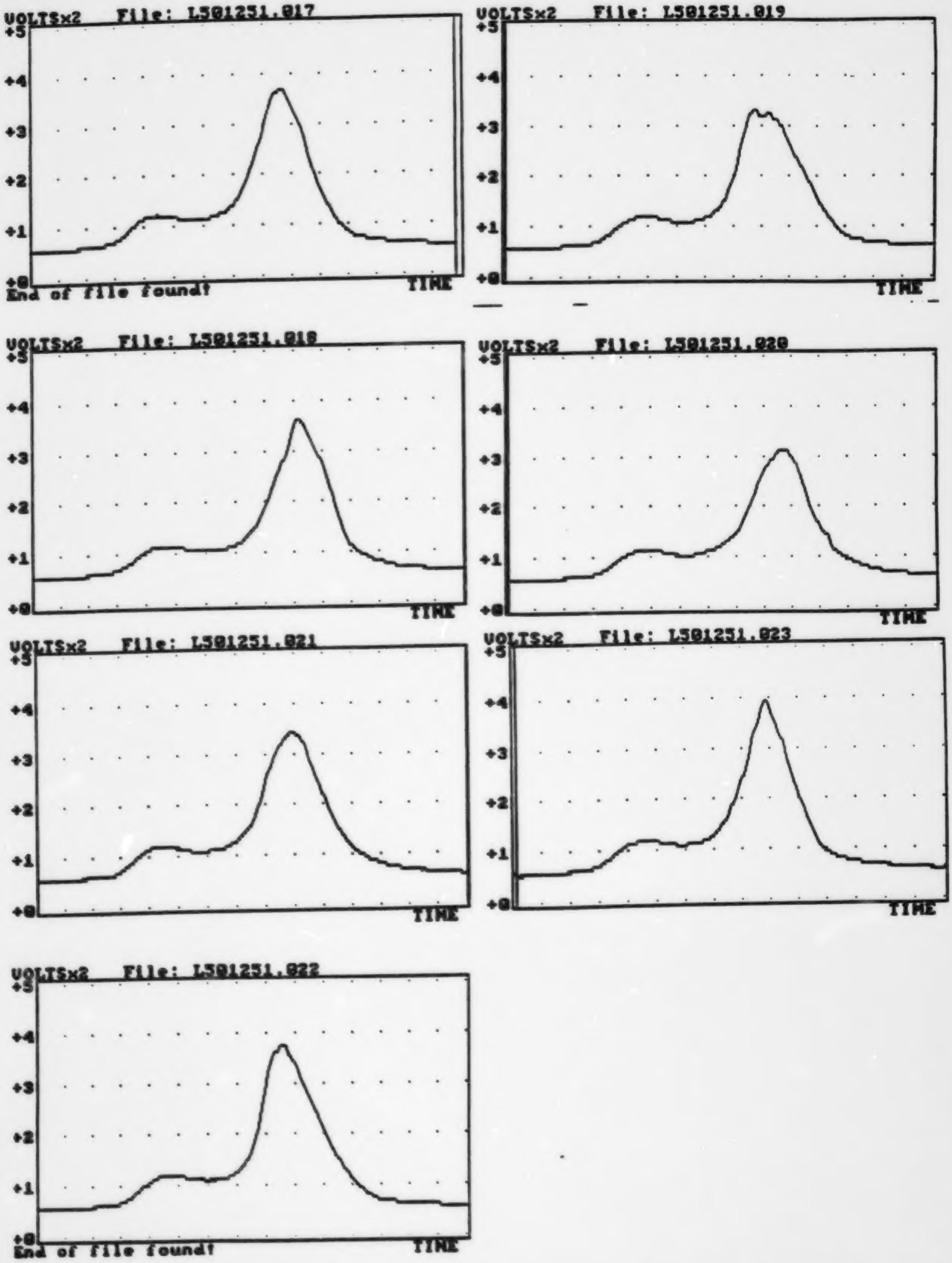


TABLE IV-7

ULTIMATE ANALYSIS OF COAL STANDARDS

Leco Standard	CANMET wt. %	Leco wt. %	
501-251	1.94	2.25	Total (t)
	0.03	0.10	Sulfate (s)
	1.19	1.42	pyrite (p)
	0.72	0.73	organic (o)
501-256	2.65	2.91	t
	0.61	0.73	s
	0.74	1.00	p
	1.30	1.18	o
501-531	3.84	4.11	t
	0.57	0.66	s
	1.44	2.09	p
	1.83	1.36	o
501-648	0.41	0.51	t
	0.00	0.06	s
	0.10	0.03	p
	0.31	0.42	o
501-701	0.97	1.11	t
	0.00	0.02	s
	0.33	0.38	p
	0.64	0.71	o

TABLE IV-8

CALCULATION OF K FACTOR

FILE	AREA1	Total AREA	Leco Apyr	Leco K	Canmet Apyr	Canmet K
A701.121	11927	67308	23465	0.5083	22899	0.5209
.122	7336	64489	22482	0.3263	21940	.3344
.124	14449	61538	21454	0.6735	20936	.6902
			mean =	0.5027	mean =	.5152
A648.015	3832	23412	1561	2.4548	5710	0.6711
.016	7988	38988	2599	3.0735	9509	0.8400
.017	9494	40743	2716	3.4956	9937	0.9554
.018	7093	34468	2298	3.0866	8407	0.8437
.019	11921	41002	2733	4.3619	10000	1.1921
			mean =	3.2945	mean =	0.9005
A251.033	18292	88940	58742	0.3114	55413	0.3301
.034	23416	88364	58361	0.4012	55054	0.4253
.035	20240	90537	59796	0.3385	56408	0.3588
.036	23875	87248	57624	0.4143	54359	0.4392
.037	23181	88339	58345	0.3973	55038	0.4212
			mean =	0.3725	mean =	0.3949
A256.047	31009	122443	56166	0.5521	44416	0.6981
.048	21650	107096	49127	0.4407	38848	0.5573
.049	29489	118708	54453	0.5415	43061	0.5848
.050	22657	112821	51753	0.4378	40925	0.5536
.051	22350	109738	50330	0.4440	39807	0.5615
			mean =	0.4832	mean =	0.6111
A531.014	36325	140612	85182	0.4264	61921	0.5866
.015	37031	121737	73748	0.5021	53609	0.6908
.016	35671	122255	74062	0.4816	53837	0.6626
.017	43742	141259	85574	0.5112	62206	0.7032
.018	37362	133665	80974	0.4614	58862	0.6347
			mean =	0.4765	mean =	0.6556

TABLE IV-9
RESULTS OF DECON PROGRAM

	Leco			Canmet	
FILE	% ORG	%PYR		% ORG	%PYR
A701.121	1.01	0.55		1.02	0.54
.122	1.15	0.34		1.16	0.33
.124	0.78	0.68		0.80	0.67
Leco "True"	0.71	0.38	Canmet "True"	0.64	0.33
	K = 0.5027			K = 0.5152	
A648.015	0.53	0.03		0.46	0.10
.016	0.92	0.06		0.76	0.22
.017	0.95	0.07		0.75	0.26
.018	0.78	0.05		0.65	0.19
.019	0.93	0.09		0.69	0.33
Leco "True"	0.42	0.03	Canmet "True"	0.31	0.10
	K = 3.2945			K = 0.9005	
A251.033	0.99	1.22		1.05	1.15
.034	0.64	1.57		0.73	1.48
.035	0.84	1.26		0.91	1.19
.036	0.57	1.59		0.66	1.50
.037	0.65	1.54		0.73	1.45
Leco "True"	0.73	1.42	Canmet "True"	0.72	1.19
	K = 0.3725			K = 0.3949	
A256.047	1.46	1.60		1.79	1.27
.048	1.56	1.12		1.79	0.89
.049	1.39	1.47		1.69	1.16
.050	1.65	1.17		1.89	0.93
.051	1.60	1.17		1.85	0.92
Leco "True"	1.18	1.00	Canmet "True"	1.30	0.74
	K = 0.4832			K = 0.6111	
A531.014	1.59	1.89		2.11	1.37
.015	1.08	1.90		1.60	1.38
.016	1.14	1.80		1.63	1.31
.017	1.21	2.25		1.83	1.64
.018	1.37	1.94		1.90	1.41
Leco "True"	1.36	12.09	Canmet "True"	1.83	1.44
	K = 0.4765			K = 0.6556	

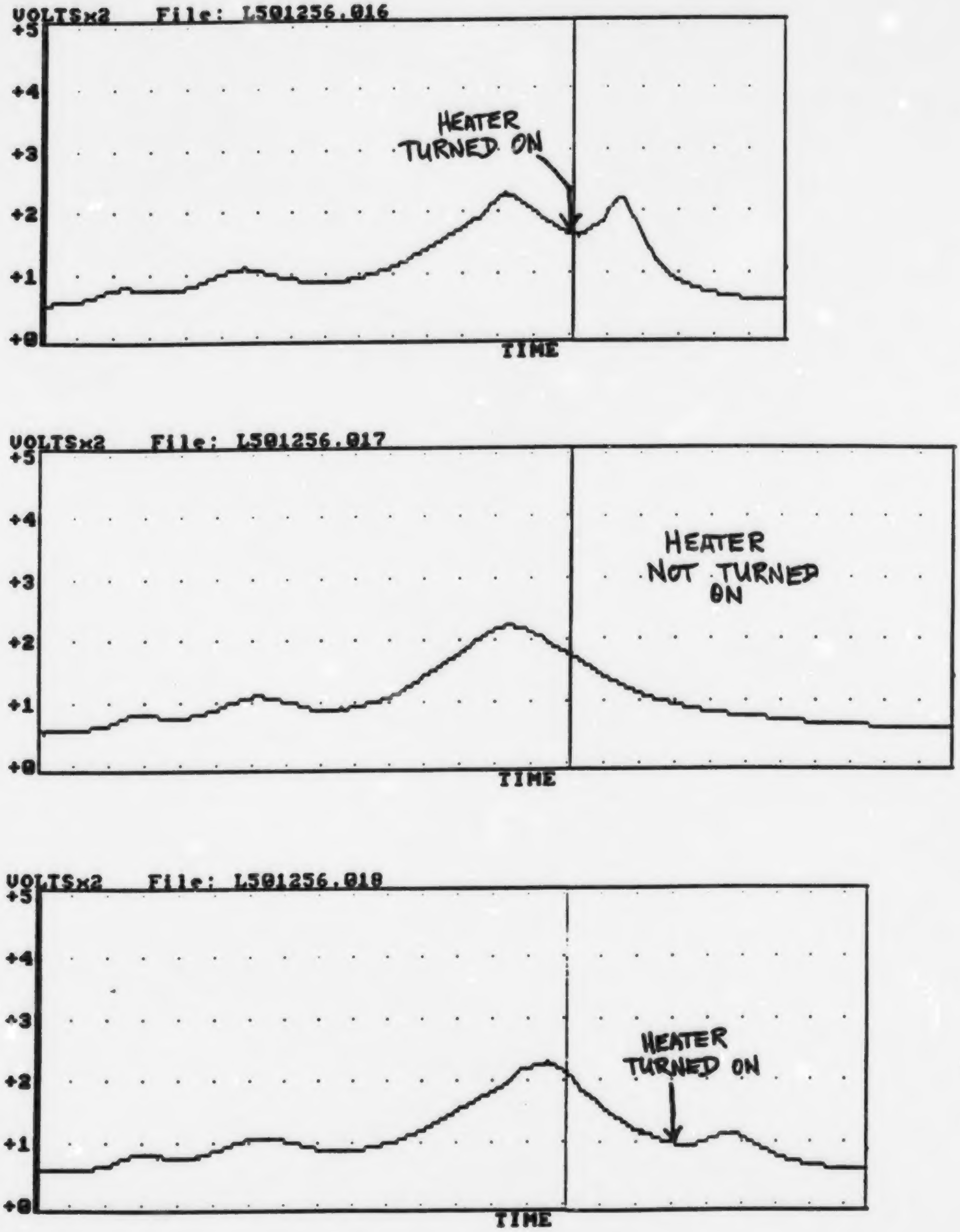
IV.7 LOCAL HEATER STUDIES

One approach that was taken to improve the final results was the implementation of a local heater to provide rapid temperature ramping of the boat within the furnace. When this local heater was installed and tested, a short circuit developed in the heater. The design of the heater was modified so as to prevent the recurrence of the short circuit and a second version of the heater was fabricated. Tests of the heater indicated that when the furnace is at 550°C, the heater can take the sample boat temperature up to 900°C within 3 minutes at a setting of 75 volts. More rapid heating than this is not desirable.

Spectrochronograms were acquired for Leco Standard 501-256 (1.18% organic, 1.00% pyritic, and 0.73% sulfate) with the local heater installed. Examples are shown in Figure IV-36. File L501256.017 was run with the heater in place but not turned on. A comparison of this spectrochronogram with similar ones that do not have the heater installed shows the effect of just physically having the heater in place. The boat sits above the local heater so that its thermal environment and gas flows around it are somewhat changed. The most obvious effect is that all of the observed peaks are shifted to a later time and broadened. In file L501256.016, the local heater was switched on at 5 minutes after the start. In file L501256.018, the heater was turned on at 6 minutes after the start. The purpose of turning the heater on this late in the run is to resolve the sulfate from the tail of the pyrite peak. in L5001256.016, the sharp peak after the heater was turned on was probably due to the release of sulfate as well as the more rapid evolution of the remaining pyritic S.

FIGURE IV-36

SPECTROCHRONOGRAMS WITH LOCAL HEATER INSTALLED



IV.8 SYNTHETIC SPECTROCHRONOGRAMS - FURTHER STUDIES

Studies were done using synthetic spectrochronograms to test the efficacy of the deconvolution program as described in Section IV.3. Later in the program, more synthetic chronograms were generated and submitted to the deconvolution program for testing.

The first new series of synthetic spectrochronograms had two Lorentzian peaks. The true values for the peak parameters are given in Table IV-10.

The spectrochronograms and the results of the deconvolution program are shown in Figures IV-37 to IV-51. For files FAKE17.001 through FAKE31.001, the correct results were always found. FAKE31.001 was a test of a case with extreme overlap. It is not obvious from looking at the spectrochronogram that there is even a second peak present. However, even in this case the correct peaks were found.

One version of the deconvolution software was modified so that when it is searching for three peaks, it asks the operator to estimate the three peak positions. Three synthetic spectrochronograms were created each having three peaks. They had the true values as shown in Table IV-11. The chronograms are shown in Figures IV-52 to IV-54.

In this case, because the operator provides the initial guess, the success of the deconvolution depends on the accuracy of the operator's estimates.

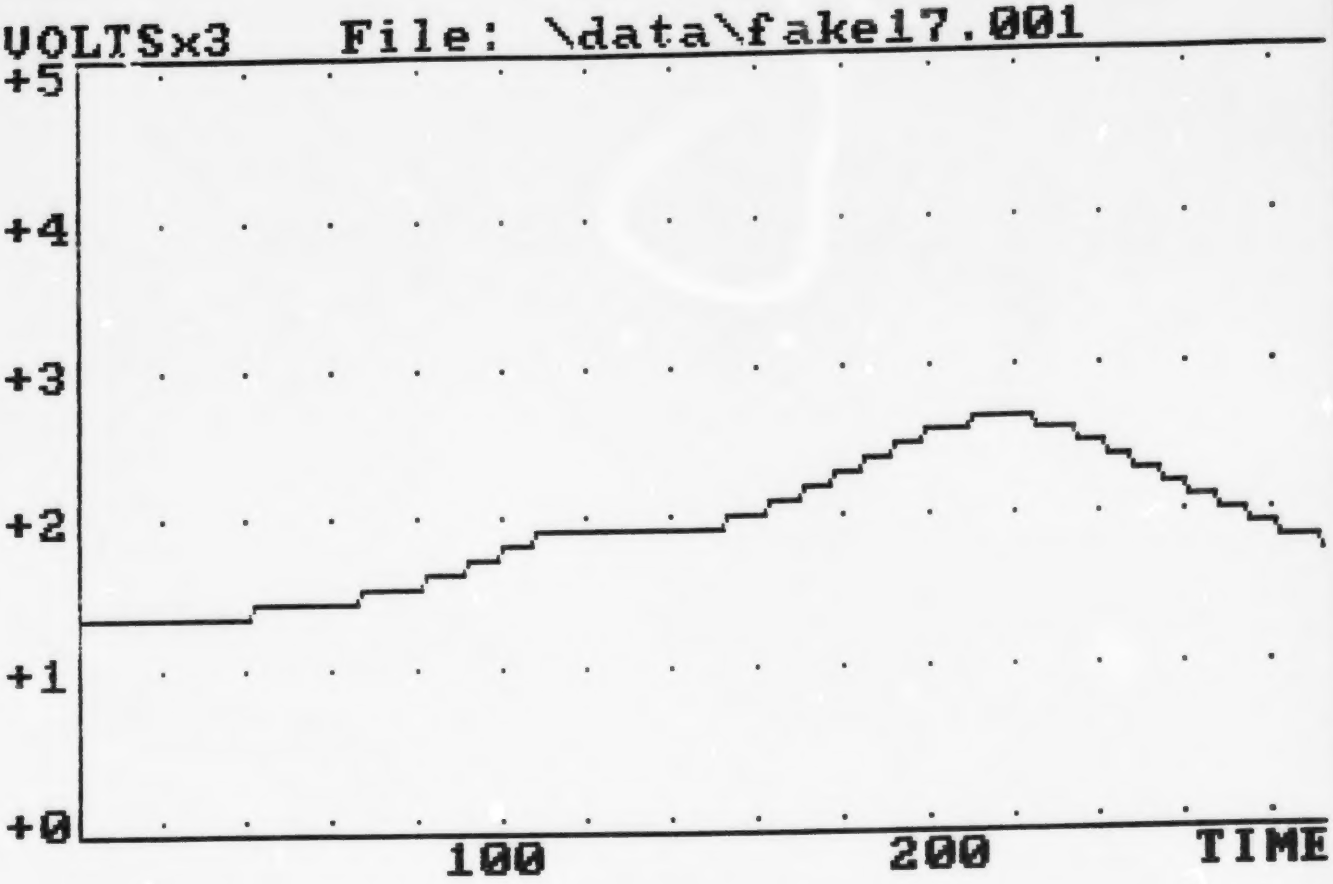
Figures IV-55 through IV-59 show the output of the DECON program for these synthetic chronograms using various initial guesses for the peak positions.

TABLE IV-10

TRUE PARAMETERS FOR SYNTHETIC SPECTROCHRONOGRAMS

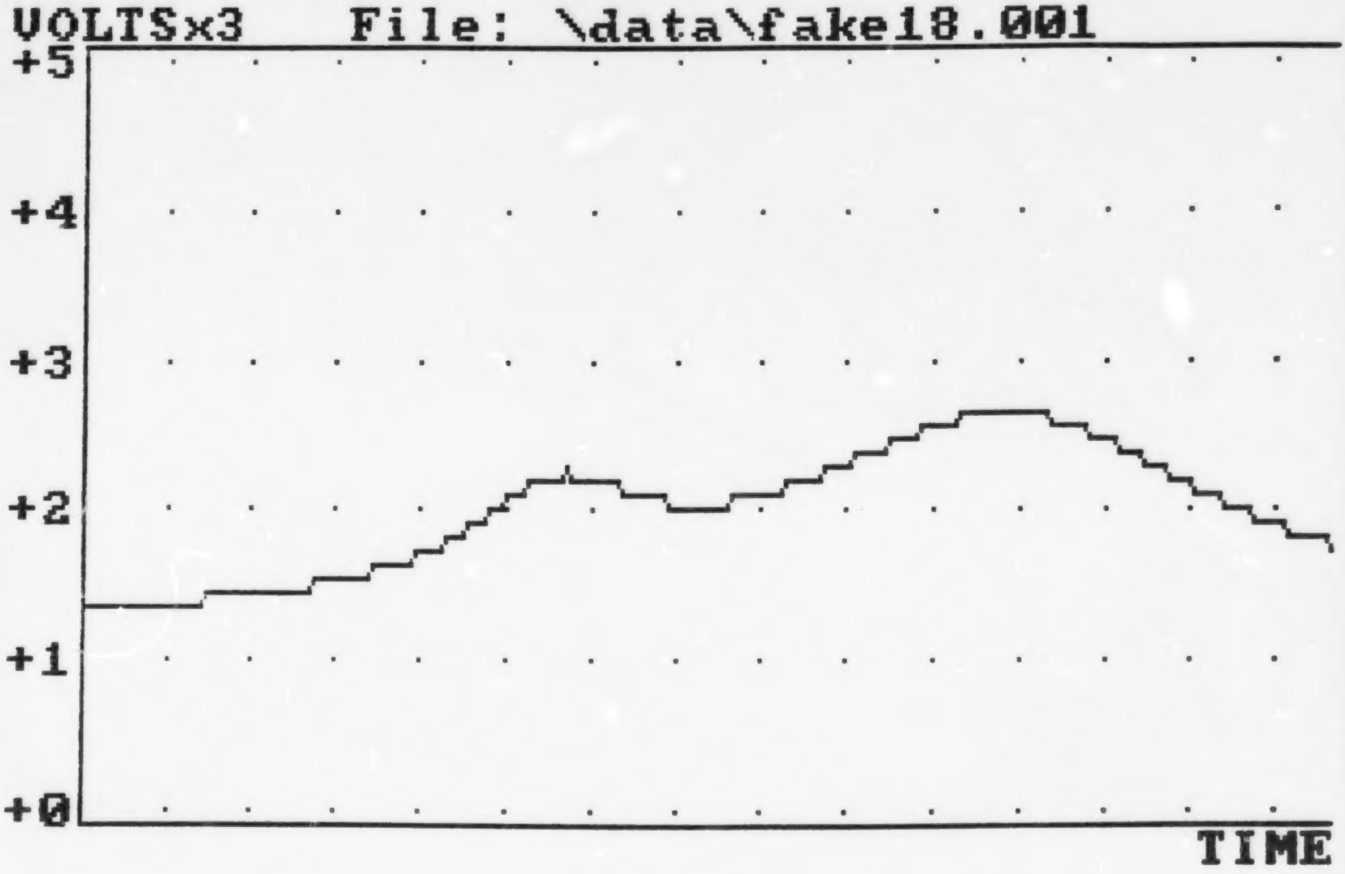
File	Mean1	Ampl1	Width1	Mean2	Ampl2	Width2	Baseline
FAKE17	110	7000	50	215	70,000	120	350
FAKE18	110	14000	50	215	70,000	120	350
FAKE19	110	3000	50	215	70,000	120	350
FAKE20	110	7000	50	215	30,000	120	350
FAKE21	110	7000	50	215	140,000	120	350
FAKE22	110	7000	100	215	70,000	120	350
FAKE23	110	7000	25	215	70,000	120	350
FAKE24	110	7000	50	215	70,000	200	350
FAKE25	110	7000	50	215	70,000	50	350
FAKE26	110	14000	100	215	70,000	120	350
FAKE27	110	3000	25	215	70,000	120	350
FAKE28	110	7000	50	215	140,000	200	350
FAKE29	110	7000	50	215	30,000	50	350
FAKE30	50	7000	50	215	70,000	120	350
FAKE31	170	7000	50	215	70,000	120	350

Figure IV-37



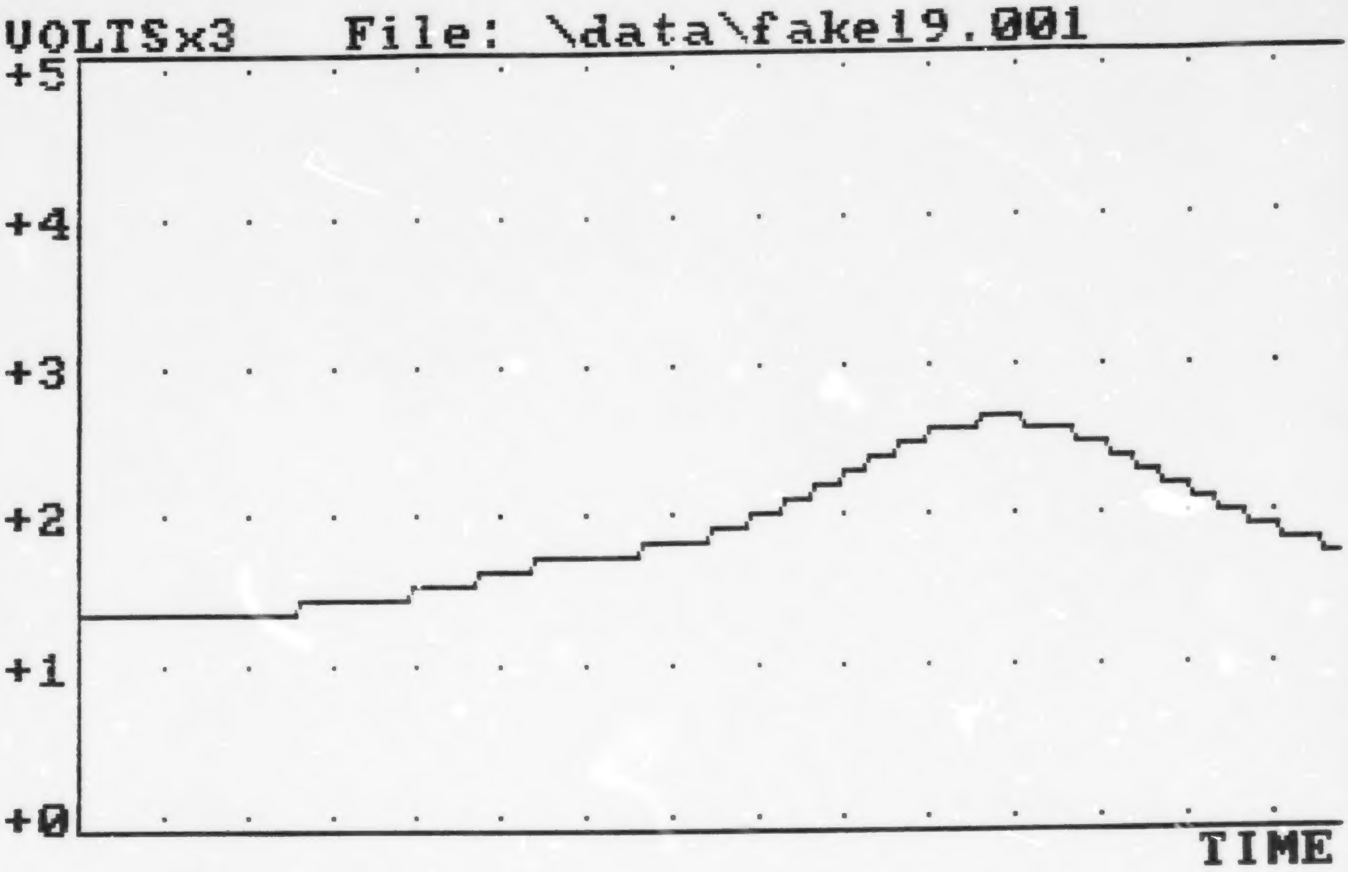
```
What´s the file name ?
File: fake17.001
NPTS IS  377
 ON THE  1 ITERATION CHISQR =    119.562  NORMALIZED  RESIDUAL =     9.802
 ON THE  2 ITERATION CHISQR =      7.325  NORMALIZED  RESIDUAL =     1.986
 ON THE  3 ITERATION CHISQR =       .398  NORMALIZED  RESIDUAL =      .460
 ON THE  4 ITERATION CHISQR =       .160  NORMALIZED  RESIDUAL =      .318
 ON THE  5 ITERATION CHISQR =       .160  NORMALIZED  RESIDUAL =      .318
                 INITIAL                      FINAL
  MEAN1          111.000                    109.966   +/-      .03
  AMPL1        11875.950                   6982.851   +/-    17.42
  WIDTH1         117.000                     50.033   +/-      .12
  MEAN2          213.000                    215.001   +/-      .01
  AMPL2        63373.051                  70107.673   +/-    41.73
  WIDTH2         117.000                    120.134   +/-      .06
  BASELINE       380.000                    348.962   +/-      .08
  GAIN              .040                     -.001   +/-      .00
 THE NUMBER OF ITERATIONS =  5
  AREA1 =     6276.     AREA2 =   56113.      .TOTAL AREA =   62389.
AREA1 is  .101 of the total area.
Peak1:  .016 %        Peak2:  .140 %
```

Figure IV-38



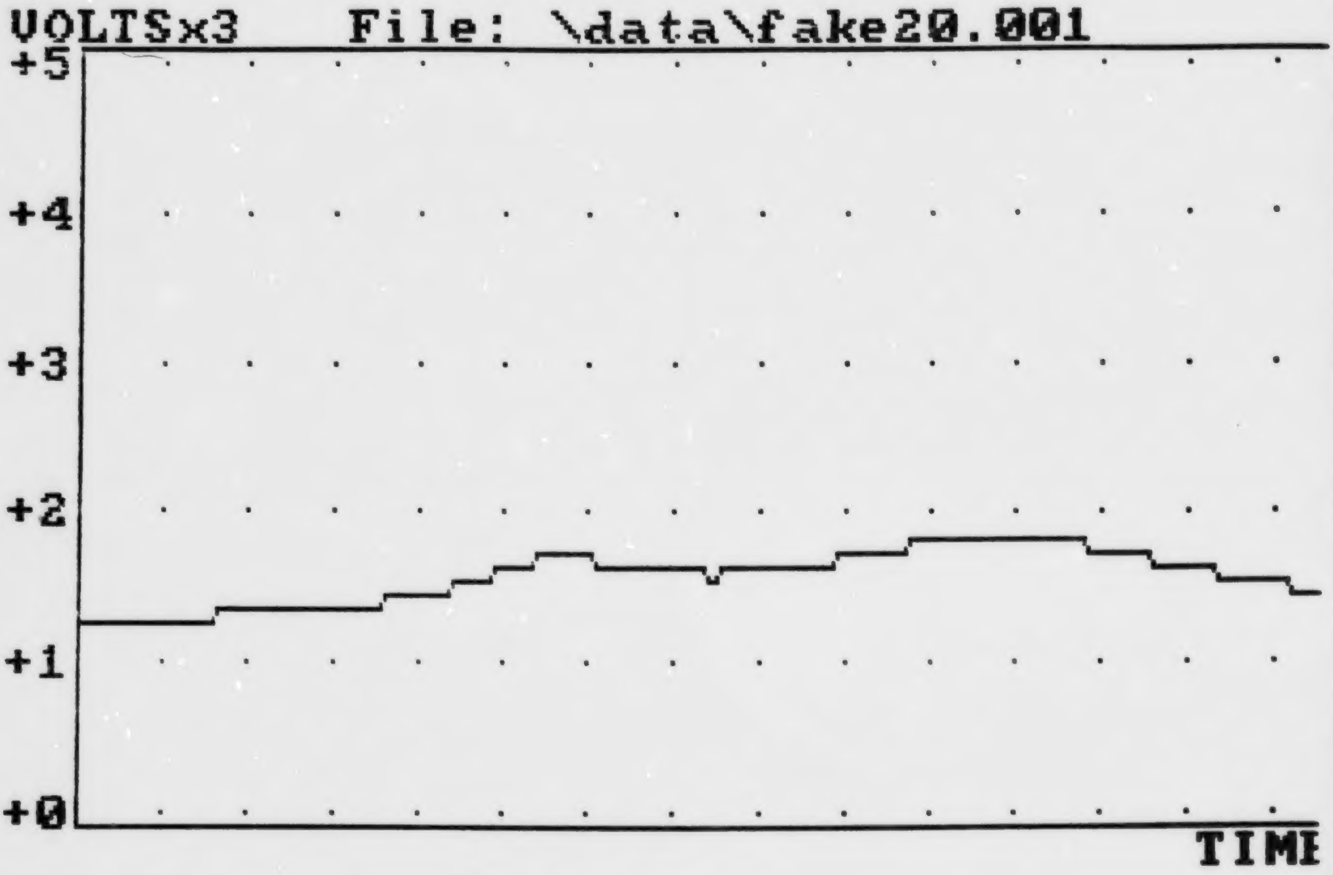
```
What's the file name ?
File: fake18.001
There are  378 data points.
 ON THE  1 ITERATION CHISQR =    6910.247  NORMALIZED  RESIDUAL =    72.149
 ON THE  2 ITERATION CHISQR =    5118.589  NORMALIZED  RESIDUAL =    65.246
 ON THE  3 ITERATION CHISQR =    1319.410  NORMALIZED  RESIDUAL =    28.422
 ON THE  4 ITERATION CHISQR =     934.025  NORMALIZED  RESIDUAL =    21.035
 ON THE  5 ITERATION CHISQR =     103.343  NORMALIZED  RESIDUAL =     6.322
 ON THE  6 ITERATION CHISQR =       4.725  NORMALIZED  RESIDUAL =     1.338
 ON THE  7 ITERATION CHISQR =        .172  NORMALIZED  RESIDUAL =      .334
 ON THE  8 ITERATION CHISQR =        .158  NORMALIZED  RESIDUAL =      .322
 ON THE  9 ITERATION CHISQR =        .158  NORMALIZED  RESIDUAL =      .322
                INITIAL                   FINAL
  MEAN1         111.000            109.984  +/-    .01
  AMPL1       24413.229          13976.519  +/-  17.28
  WIDTH1        174.000             49.959  +/-    .06
  MEAN2         213.000            215.001  +/-    .01
  AMPL2       94520.281          70097.576  +/-  41.25
  WIDTH2        174.000            120.117  +/-    .06
  BASELINE      384.000            348.993  +/-    .08
  GAIN             .029              -.001  +/-    .00
 THE NUMBER OF ITERATIONS =  9
  AREA1 =   12566.      AREA2 =   56151.      TOTAL AREA =   68717.
AREA1 is  .183 of the total area.
Peak1:  .031 %         Peak2:  .140 %
```

Figure IV-39



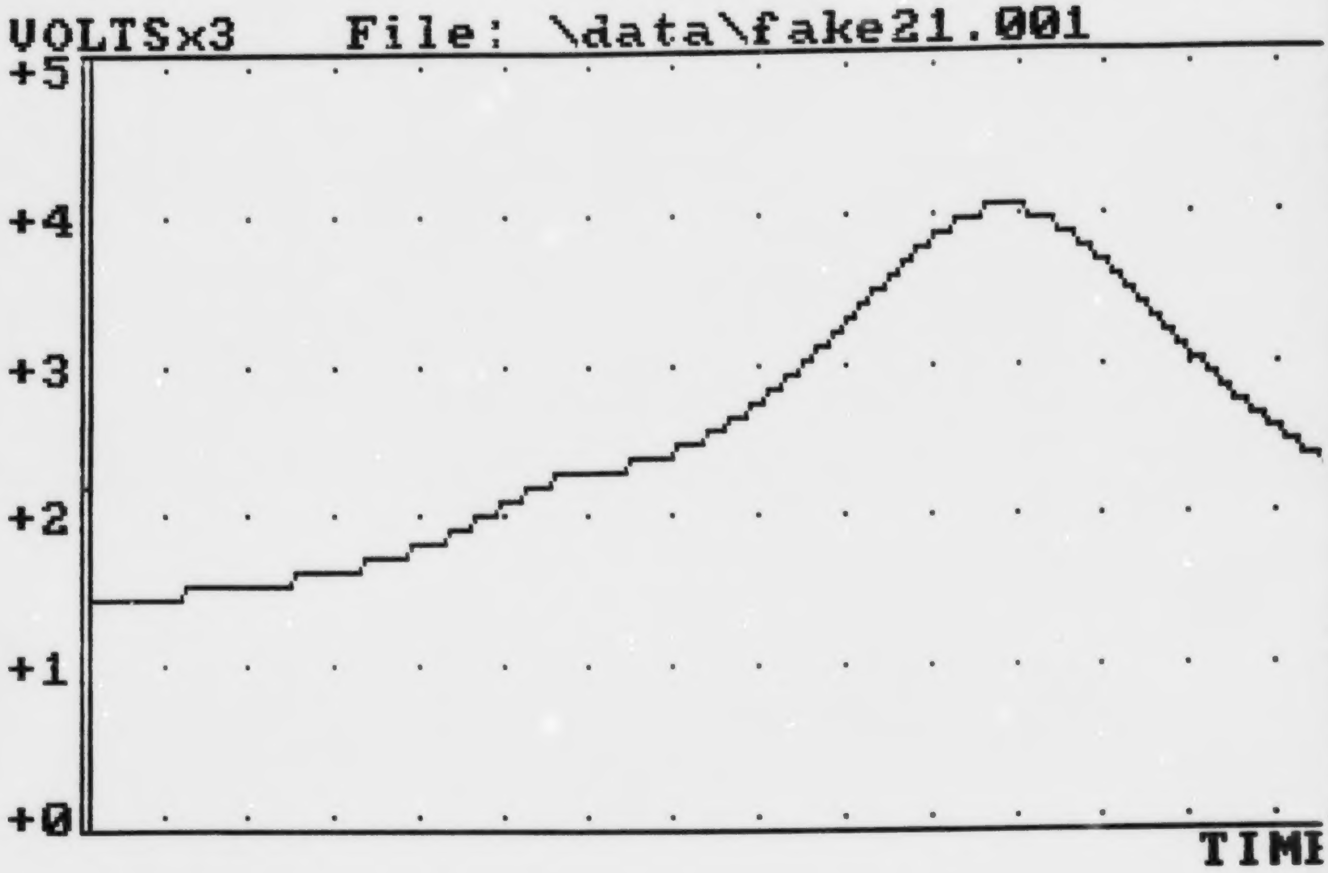
```
fake19.001
File: fake19.001
There are  372 data points.
 ON THE  1 ITERATION CHISQR =      40.129  NORMALIZED  RESIDUAL =     4.831
 ON THE  2 ITERATION CHISQR =       2.466  NORMALIZED  RESIDUAL =      .894
 ON THE  3 ITERATION CHISQR =        .260  NORMALIZED  RESIDUAL =      .394
 ON THE  4 ITERATION CHISQR =        .171  NORMALIZED  RESIDUAL =      .334
 ON THE  5 ITERATION CHISQR =        .171  NORMALIZED  RESIDUAL =      .335
                INITIAL                      FINAL
  MEAN1          110.000             109.977  +/-     .07
  AMPL1         4143.088            3014.070  +/-   18.01
  WIDTH1         113.000              50.055  +/-     .28
  MEAN2          215.000             214.998  +/-     .01
  AMPL2        61383.863           69973.239  +/-   44.25
  WIDTH2         113.000             119.940  +/-     .06
  BASELINE       377.000             349.018  +/-     .08
  GAIN              .054                .000  +/-     .00
 THE NUMBER OF ITERATIONS =  5
  AREA1 =     2707.     AREA2 =   55798.       TOTAL AREA =   58505.
AREA1 is  .046 of the total area.
Peak1:  .007 %        Peak2:  .139 %


C:\SULF>
```

Figure IV-40



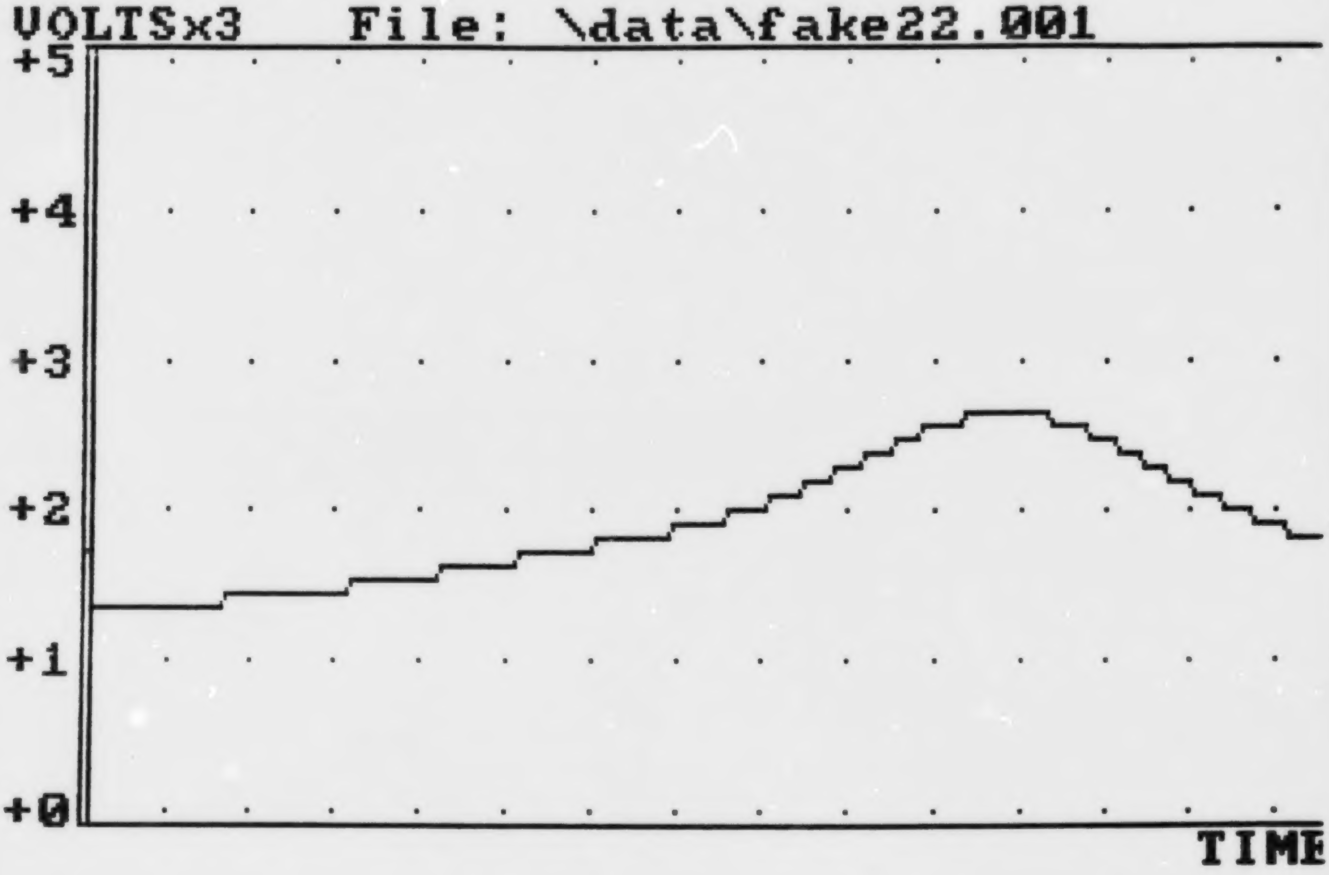
```
What's the file name ?
File: fake20.001
There are  366 data points.
 ON THE  1 ITERATION CHISQR =   1184.589  NORMALIZED  RESIDUAL =    27.901
 ON THE  2 ITERATION CHISQR =   1081.965  NORMALIZED  RESIDUAL =    23.255
 ON THE  3 ITERATION CHISQR =    540.577  NORMALIZED  RESIDUAL =    19.004
 ON THE  4 ITERATION CHISQR =    490.836  NORMALIZED  RESIDUAL =    16.888
 ON THE  5 ITERATION CHISQR =    246.208  NORMALIZED  RESIDUAL =    12.246
 ON THE  6 ITERATION CHISQR =    186.485  NORMALIZED  RESIDUAL =     9.243
 ON THE  7 ITERATION CHISQR =    147.751  NORMALIZED  RESIDUAL =     8.637
 ON THE  8 ITERATION CHISQR =    104.481  NORMALIZED  RESIDUAL =     6.939
 ON THE  9 ITERATION CHISQR =     62.234  NORMALIZED  RESIDUAL =     5.753
 ON THE 10 ITERATION CHISQR =     36.185  NORMALIZED  RESIDUAL =     4.056
 ON THE 11 ITERATION CHISQR =      6.176  NORMALIZED  RESIDUAL =     1.727
 ON THE 12 ITERATION CHISQR =       .256  NORMALIZED  RESIDUAL =      .395
 ON THE 13 ITERATION CHISQR =       .172  NORMALIZED  RESIDUAL =      .331
 ON THE 14 ITERATION CHISQR =       .172  NORMALIZED  RESIDUAL =      .331
                  INITIAL                     FINAL
  MEAN1           111.000                  215.013  +/-    .03
  AMPL1         13385.827                30044.925  +/-  46.31
  WIDTH1          176.000                  120.145  +/-    .16
  MEAN2           211.000                  110.014  +/-    .03
  AMPL2         40895.363                 6994.902  +/-  18.17
  WIDTH2          176.000                   50.057  +/-    .12
  BASELINE        365.000                  348.979  +/-    .08
  GAIN               .016                    -.000  +/-    .00
 THE NUMBER OF ITERATIONS = 14
  AREA1 =    23823.      AREA2 =     6278.       TOTAL AREA =    30101.
AREA1 is  .791 of the total area.
```

Figure IV-41



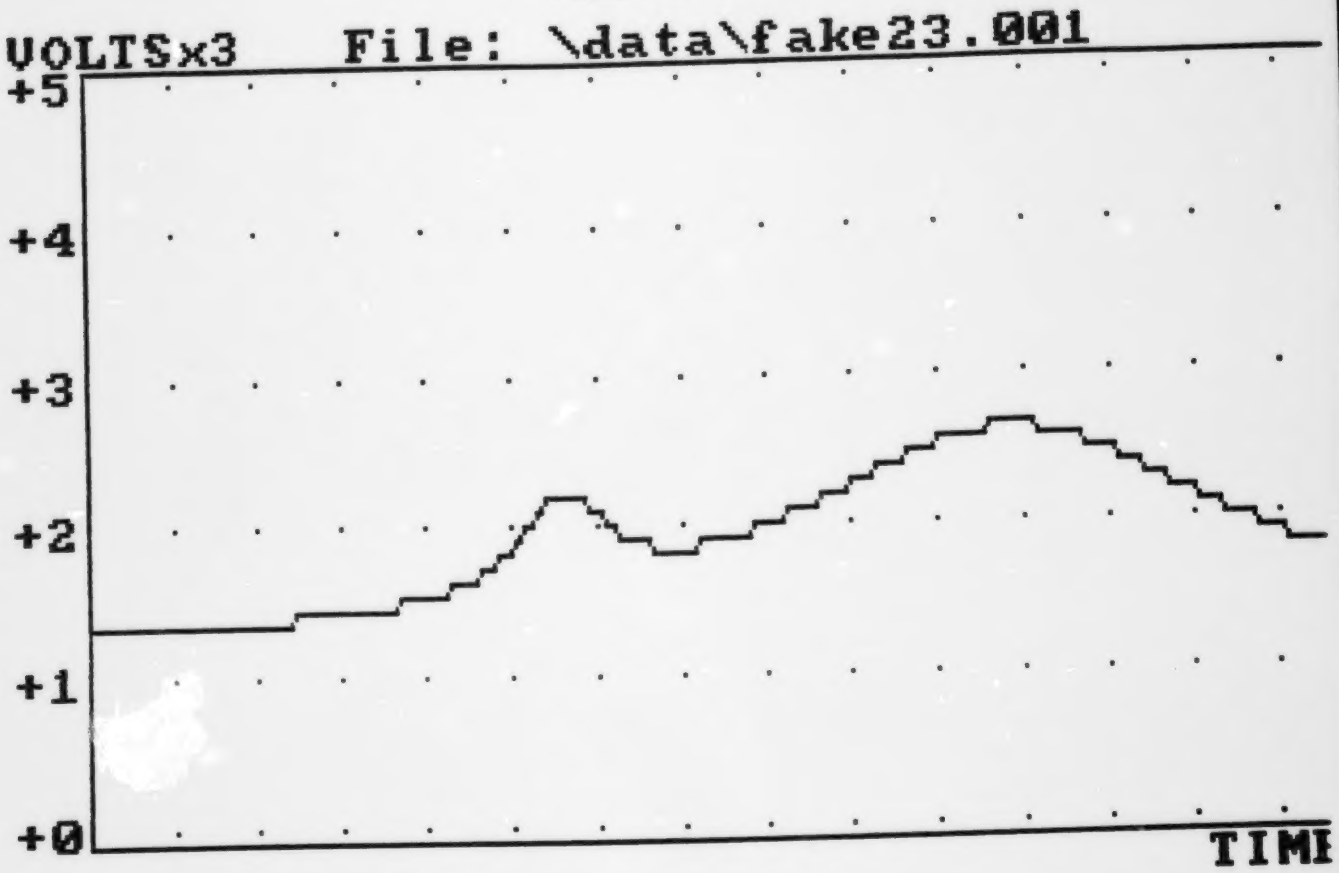
```
What's the file name ?
File: fake21.001
There are  377 data points.
 ON THE   1 ITERATION CHISQR =      191.772  NORMALIZED  RESIDUAL =    11.901
 ON THE   2 ITERATION CHISQR =       53.459  NORMALIZED  RESIDUAL =     4.237
 ON THE   3 ITERATION CHISQR =        8.971  NORMALIZED  RESIDUAL =     1.730
 ON THE   4 ITERATION CHISQR =         .381  NORMALIZED  RESIDUAL =      .455
 ON THE   5 ITERATION CHISQR =         .151  NORMALIZED  RESIDUAL =      .315
 ON THE   6 ITERATION CHISQR =         .151  NORMALIZED  RESIDUAL =      .314
 ON THE   7 ITERATION CHISQR =         .151  NORMALIZED  RESIDUAL =      .314
                 INITIAL                       FINAL
  MEAN1          113.000                    110.000  +/-     .03
  AMPL1         9866.645                   6984.423  +/-   16.96
  WIDTH1         114.000                     50.075  +/-     .11
  MEAN2          215.000                    215.004  +/-     .01
  AMPL2       123317.227                 140120.486  +/-   40.56
  WIDTH2         114.000                    120.079  +/-     .03
  BASELINE       407.000                    348.965  +/-     .08
  GAIN              .088                      -.001  +/-     .00
 THE NUMBER OF ITERATIONS =  7
  AREA1 =     6277.      AREA2 =  112162.       TOTAL AREA =  118440.
AREA1 is  .053 of the total area.
Peak1:  .016 %        Peak2:  .280 %
```

Figure IV-42



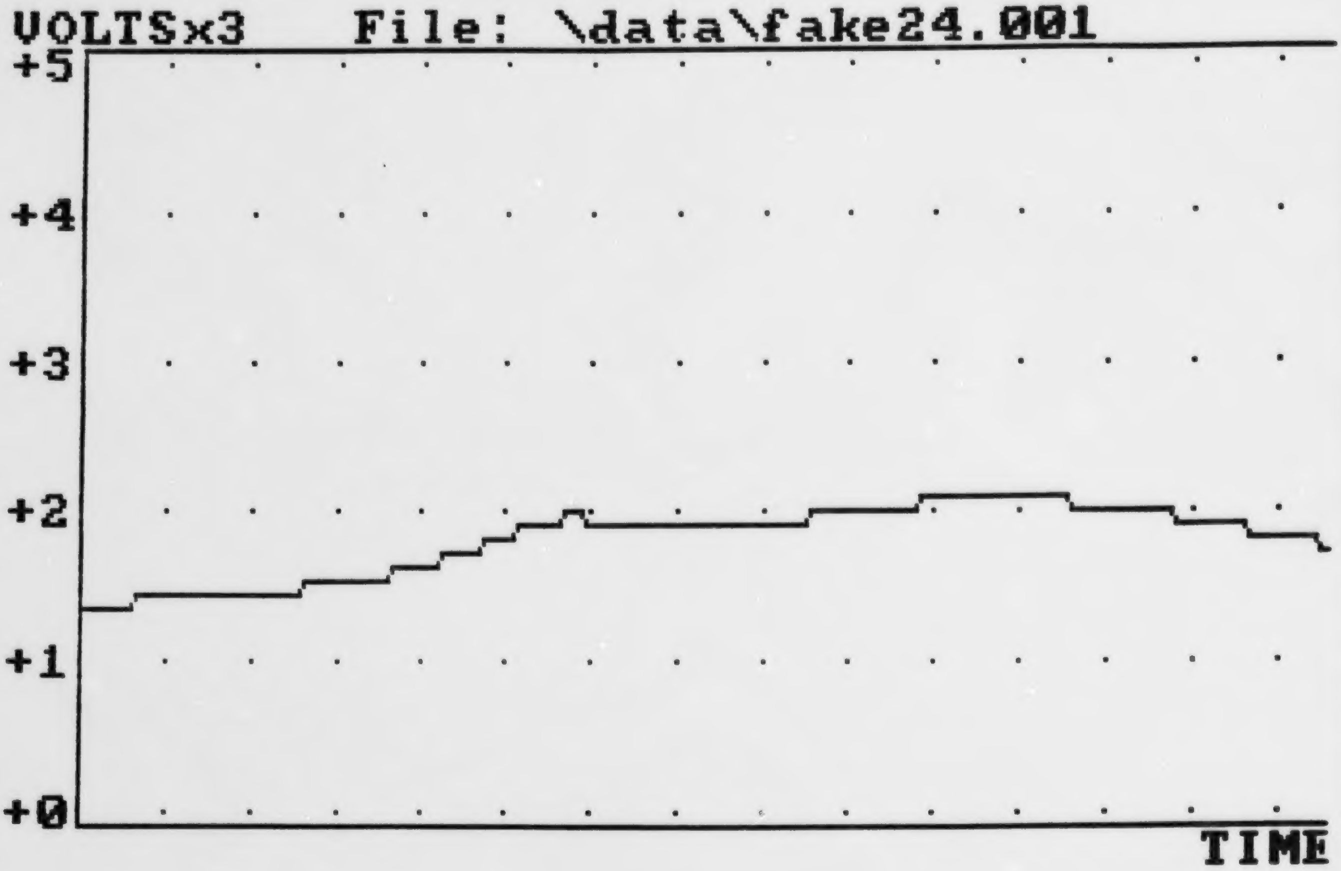
```
What´s the file name ?
File: fake22.001
There are  371 data points.
 ON THE  1 ITERATION CHISQR =     29.617  NORMALIZED  RESIDUAL =    3.397
 ON THE  2 ITERATION CHISQR =      1.423  NORMALIZED  RESIDUAL =     .822
 ON THE  3 ITERATION CHISQR =       .214  NORMALIZED  RESIDUAL =     .369
 ON THE  4 ITERATION CHISQR =       .172  NORMALIZED  RESIDUAL =     .339
 ON THE  5 ITERATION CHISQR =       .172  NORMALIZED  RESIDUAL =     .339
                 INITIAL                  FINAL
  MEAN1          116.000              109.818  +/-    .13
  AMPL1         3758.148             6936.080  +/-  61.17
  WIDTH1         116.000               99.527  +/-    .62
  MEAN2          215.000              214.987  +/-    .02
  AMPL2        63013.523            70147.823  +/-  60.59
  WIDTH2         116.000              120.176  +/-    .08
  BASELINE       383.000              348.985  +/-    .15
  GAIN              .043                -.001  +/-    .00
 THE NUMBER OF ITERATIONS =  5
  AREA1 =    5578.      AREA2 =    55864.      TOTAL AREA =   61442.
AREA1 is  .091 of the total area.
```

Figure IV-43



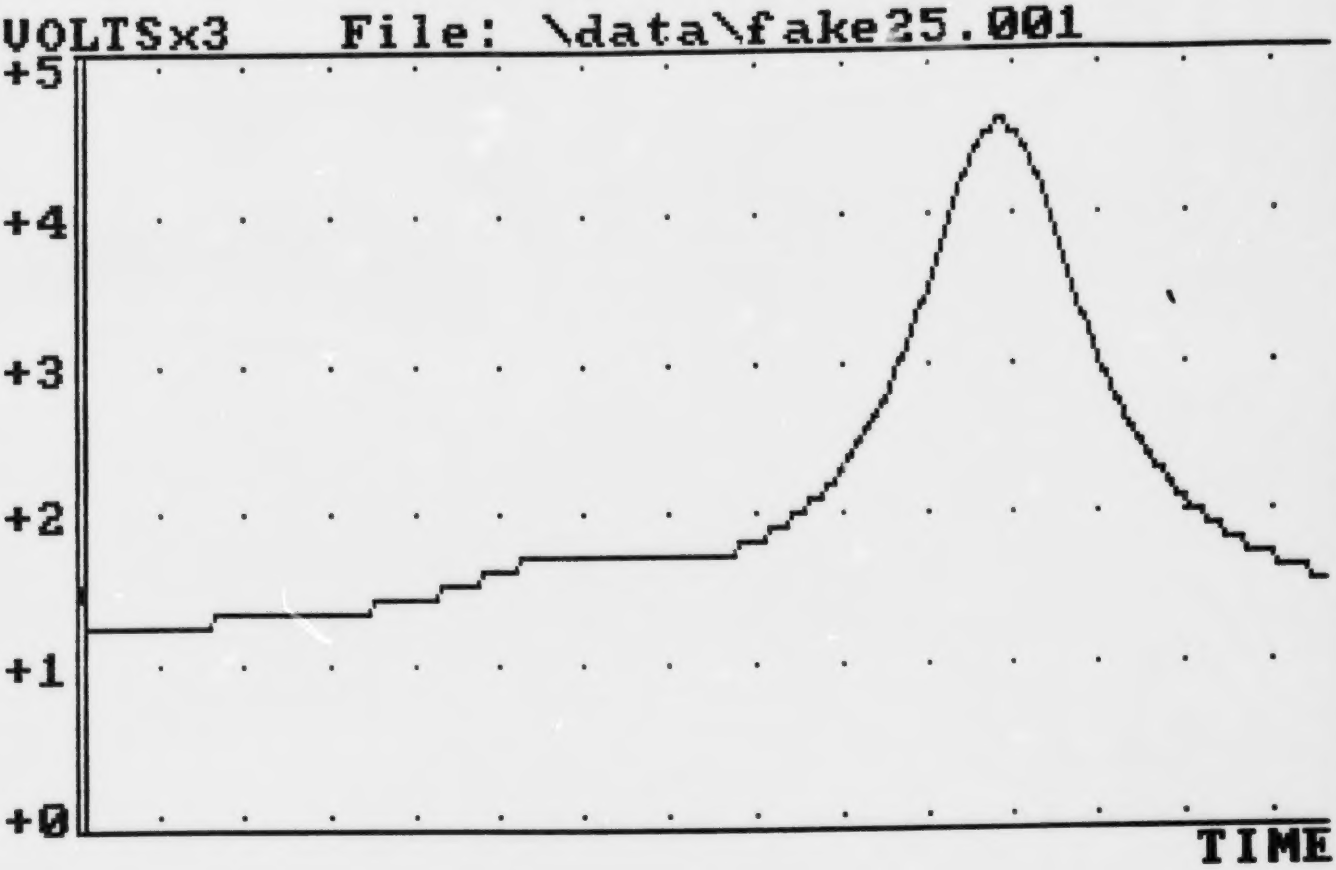
```
What´s the file name ?
File: fake23.001
There are  387 data points.
 ON THE  1 ITERATION CHISQR =    3569.894  NORMALIZED  RESIDUAL =    26.199
 ON THE  2 ITERATION CHISQR =    1543.900  NORMALIZED  RESIDUAL =    15.601
 ON THE  3 ITERATION CHISQR =    1219.597  NORMALIZED  RESIDUAL =    13.811
 ON THE  4 ITERATION CHISQR =    1176.412  NORMALIZED  RESIDUAL =    13.622
 ON THE  5 ITERATION CHISQR =     829.143  NORMALIZED  RESIDUAL =    12.278
 ON THE  6 ITERATION CHISQR =     590.482  NORMALIZED  RESIDUAL =    12.863
 ON THE  7 ITERATION CHISQR =     231.424  NORMALIZED  RESIDUAL =    10.012
 ON THE  8 ITERATION CHISQR =      32.298  NORMALIZED  RESIDUAL =     3.895
 ON THE  9 ITERATION CHISQR =        .468  NORMALIZED  RESIDUAL =      .445
 ON THE 10 ITERATION CHISQR =        .154  NORMALIZED  RESIDUAL =      .321
 ON THE 11 ITERATION CHISQR =        .154  NORMALIZED  RESIDUAL =      .321
                  INITIAL                     FINAL
  MEAN1           111.000               109.997   +/-     .01
  AMPL1         27563.957              6996.808   +/-    7.66
  WIDTH1          169.000                25.000   +/-     .03
  MEAN2           215.000               215.000   +/-     .01
  AMPL2         91538.852             70016.543   +/-   34.94
  WIDTH2          169.000               120.001   +/-     .05
  BASELINE        378.000               349.007   +/-     .07
  GAIN               .031                 -.000   +/-     .00
 THE NUMBER OF ITERATIONS = 11
  AREA1 =    6643.      AREA2 =   56477.        TOTAL AREA =   63121.
AREA1 is  .105 of the total area.
Peak1:  .017 %       Peak2:  .141 %
```

Figure IV-44



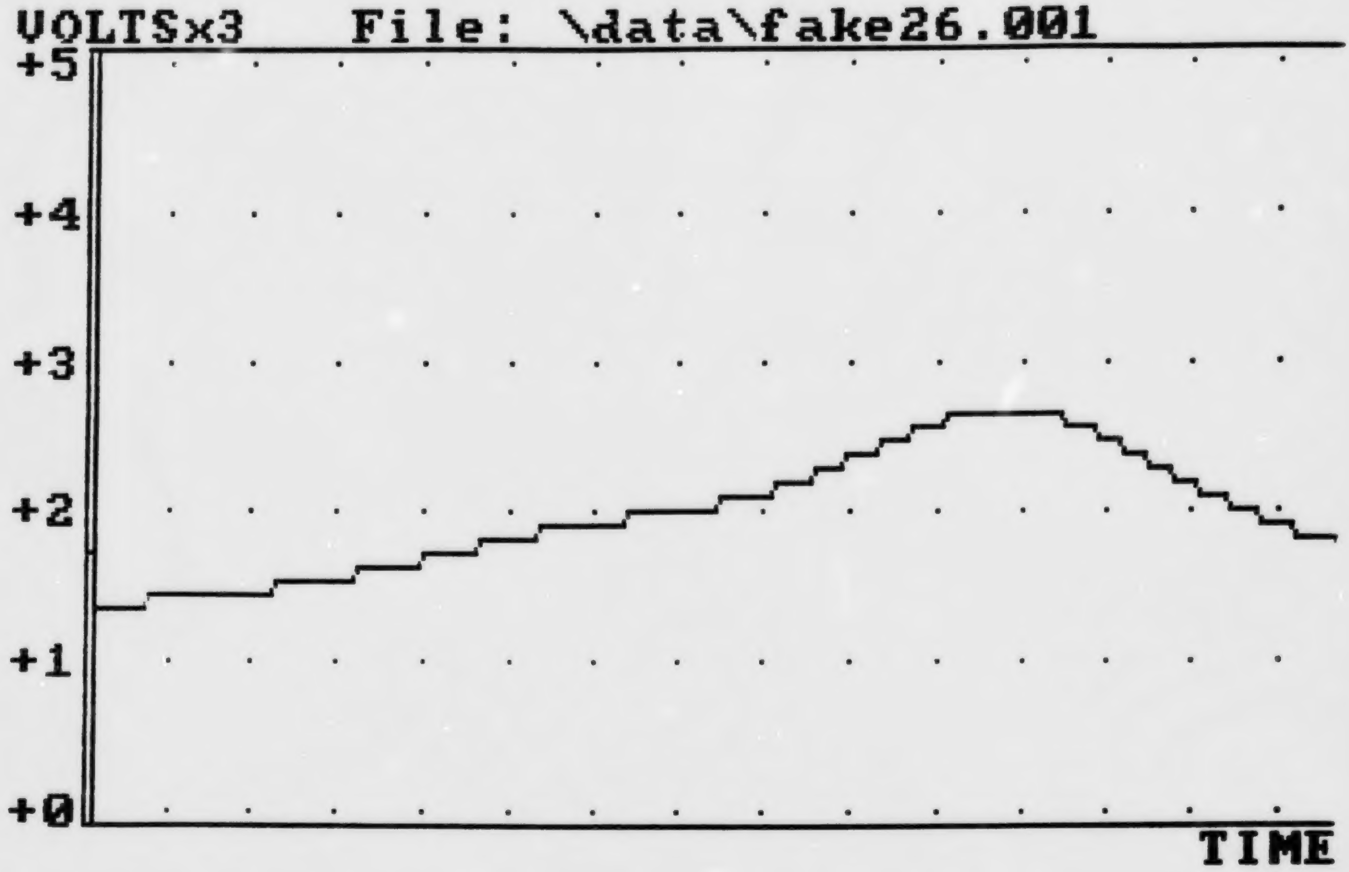
```
What´s the file name ?
File: fake24.001
There are  380 data points.
 ON THE  1 ITERATION CHISQR =    1274.804  NORMALIZED  RESIDUAL =   30.142
 ON THE  2 ITERATION CHISQR =     947.394  NORMALIZED  RESIDUAL =   26.150
 ON THE  3 ITERATION CHISQR =     651.552  NORMALIZED  RESIDUAL =   17.380
 ON THE  4 ITERATION CHISQR =     189.475  NORMALIZED  RESIDUAL =    9.837
 ON THE  5 ITERATION CHISQR =      51.440  NORMALIZED  RESIDUAL =    5.115
 ON THE  6 ITERATION CHISQR =      10.431  NORMALIZED  RESIDUAL =    2.645
 ON THE  7 ITERATION CHISQR =       3.590  NORMALIZED  RESIDUAL =    1.526
 ON THE  8 ITERATION CHISQR =        .199  NORMALIZED  RESIDUAL =     .362
 ON THE  9 ITERATION CHISQR =        .176  NORMALIZED  RESIDUAL =     .344
 ON THE 10 ITERATION CHISQR =        .176  NORMALIZED  RESIDUAL =     .344
                   INITIAL                       FINAL
  MEAN1            113.000               110.009  +/-     .03
  AMPL1          17847.400              6999.675  +/-   26.42
  WIDTH1           208.000                50.176  +/-     .15
  MEAN2            210.000               215.045  +/-     .05
  AMPL2          60087.043             70210.264  +/-  156.91
  WIDTH2           208.000               200.314  +/-     .30
  BASELINE         393.000               348.784  +/-     .15
  GAIN                .045                 -.001  +/-     .00
 THE NUMBER OF ITERATIONS = 10
  AREA1 =     6292.      AREA2 =    48284.       TOTAL AREA =    54576.
AREA1 is  .115 of the total area.
Peak1:  .016 %         Peak2:  .121 %
```

Figure IV-45



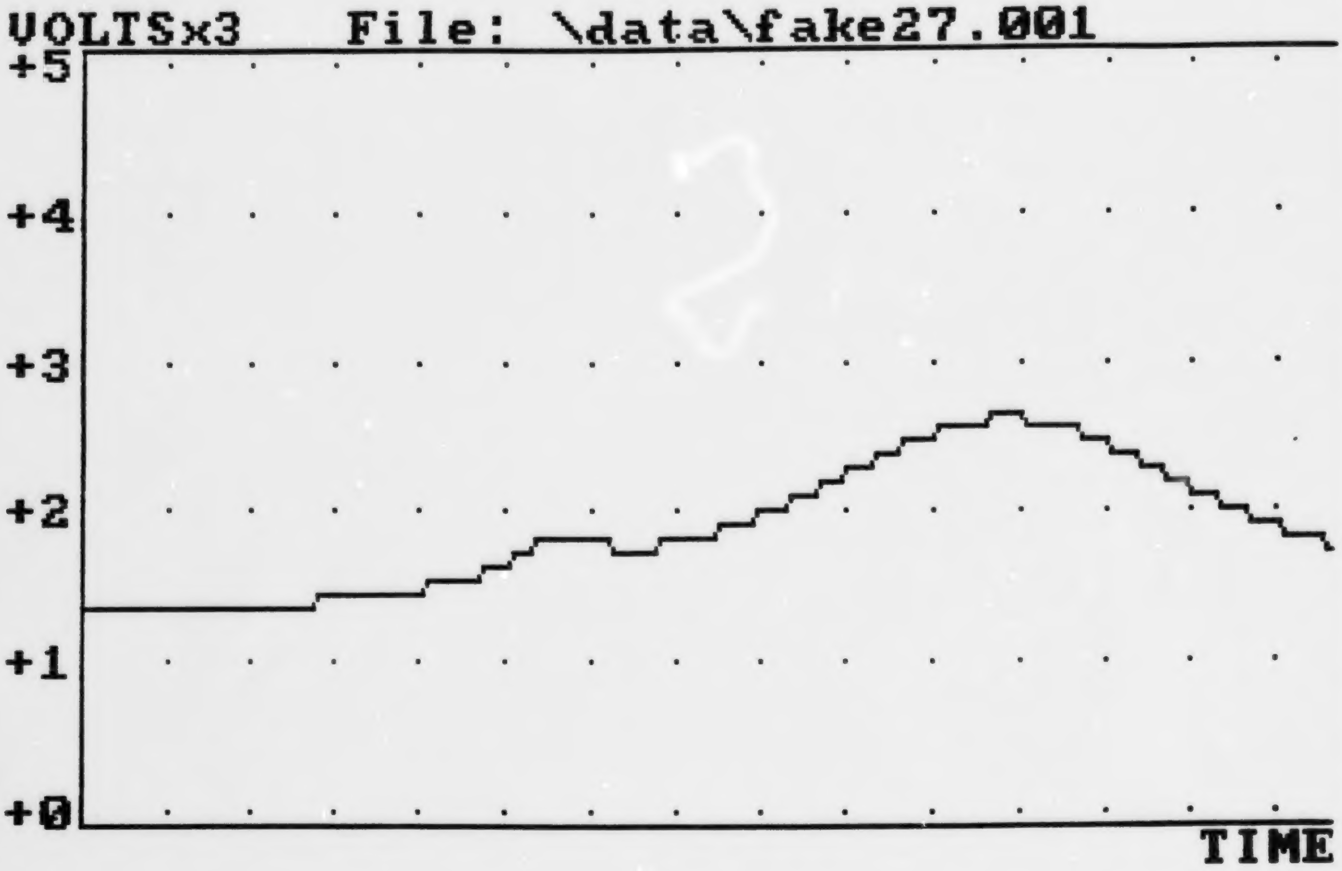
```
What´s the file name ?
fake25.001
File: fake25.001
There are  355 data points.
 ON THE  1 ITERATION CHISQR =      6.186  NORMALIZED  RESIDUAL =     1.576
 ON THE  2 ITERATION CHISQR =       .195  NORMALIZED  RESIDUAL =      .353
 ON THE  3 ITERATION CHISQR =       .194  NORMALIZED  RESIDUAL =      .354
 ON THE  4 ITERATION CHISQR =       .194  NORMALIZED  RESIDUAL =      .354
                 INITIAL                   FINAL
  MEAN1          111.000                 110.001  +/-     .03
  AMPL1         5814.918                7001.617  +/-   15.74
  WIDTH1          49.000                  50.152  +/-     .12
  MEAN2          216.000                 215.000  +/-     .00
  AMPL2        67698.398               69991.714  +/-   13.73
  WIDTH2          49.000                  49.997  +/-     .01
  BASELINE       365.000                 348.974  +/-     .08
  GAIN              .034                    .000  +/-     .00
 THE NUMBER OF ITERATIONS =  4
  AREA1 =    6273.      AREA2 =   63484.       TOTAL AREA =    69757.
AREA1 is  .090 of the total area.
Peak1:  .016 %       Peak2:  .159 %


C:\SULF>
```

Figure IV-46



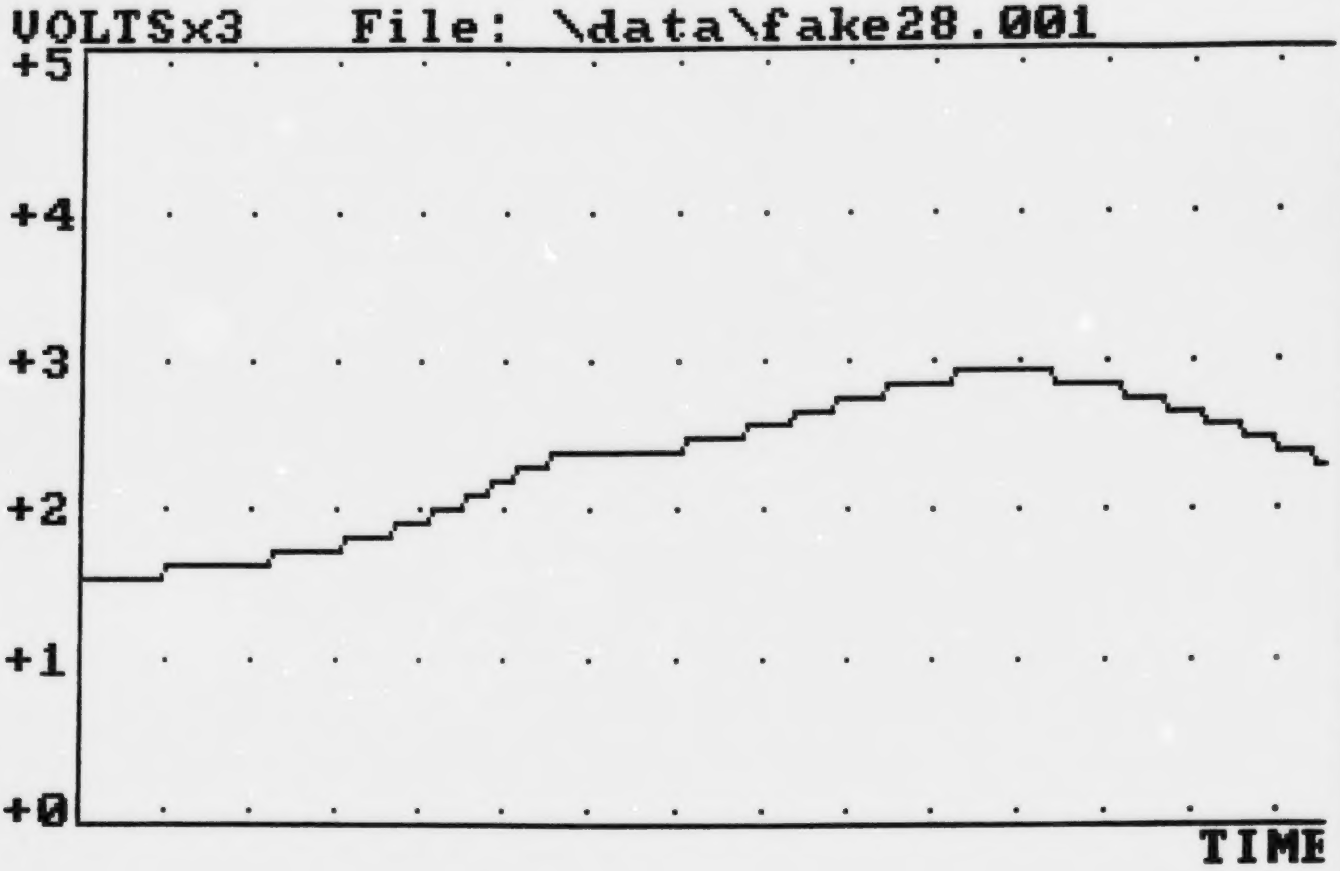
```
File: fake26.001
There are  367 data points.
 ON THE  1 ITERATION CHISQR =    251.540  NORMALIZED  RESIDUAL =    8.531
 ON THE  2 ITERATION CHISQR =     16.273  NORMALIZED  RESIDUAL =    2.442
 ON THE  3 ITERATION CHISQR =      1.039  NORMALIZED  RESIDUAL =     .759
 ON THE  4 ITERATION CHISQR =       .199  NORMALIZED  RESIDUAL =     .355
 ON THE  5 ITERATION CHISQR =       .164  NORMALIZED  RESIDUAL =     .327
 ON THE  6 ITERATION CHISQR =       .164  NORMALIZED  RESIDUAL =     .327
                 INITIAL                FINAL
  MEAN1          110.000              109.925  +/-    .06
  AMPL1         8674.186            13967.754  +/-  60.54
  WIDTH1         126.000               99.995  +/-    .31
  MEAN2          213.000              214.989  +/-    .02
  AMPL2        68445.719            70103.716  +/-  60.83
  WIDTH2         126.000              120.133  +/-    .08
  BASELINE       391.000              348.951  +/-    .15
  GAIN              .033                -.000  +/-    .00
 THE NUMBER OF ITERATIONS =  6
  AREA1 =   11210.      AREA2 =   55638.       TOTAL AREA =   66848.
AREA1 is  .168 of the total area.
Peak1:  .028 %       Peak2:  .139 %


C:\SULF>
```

Figure IV-47



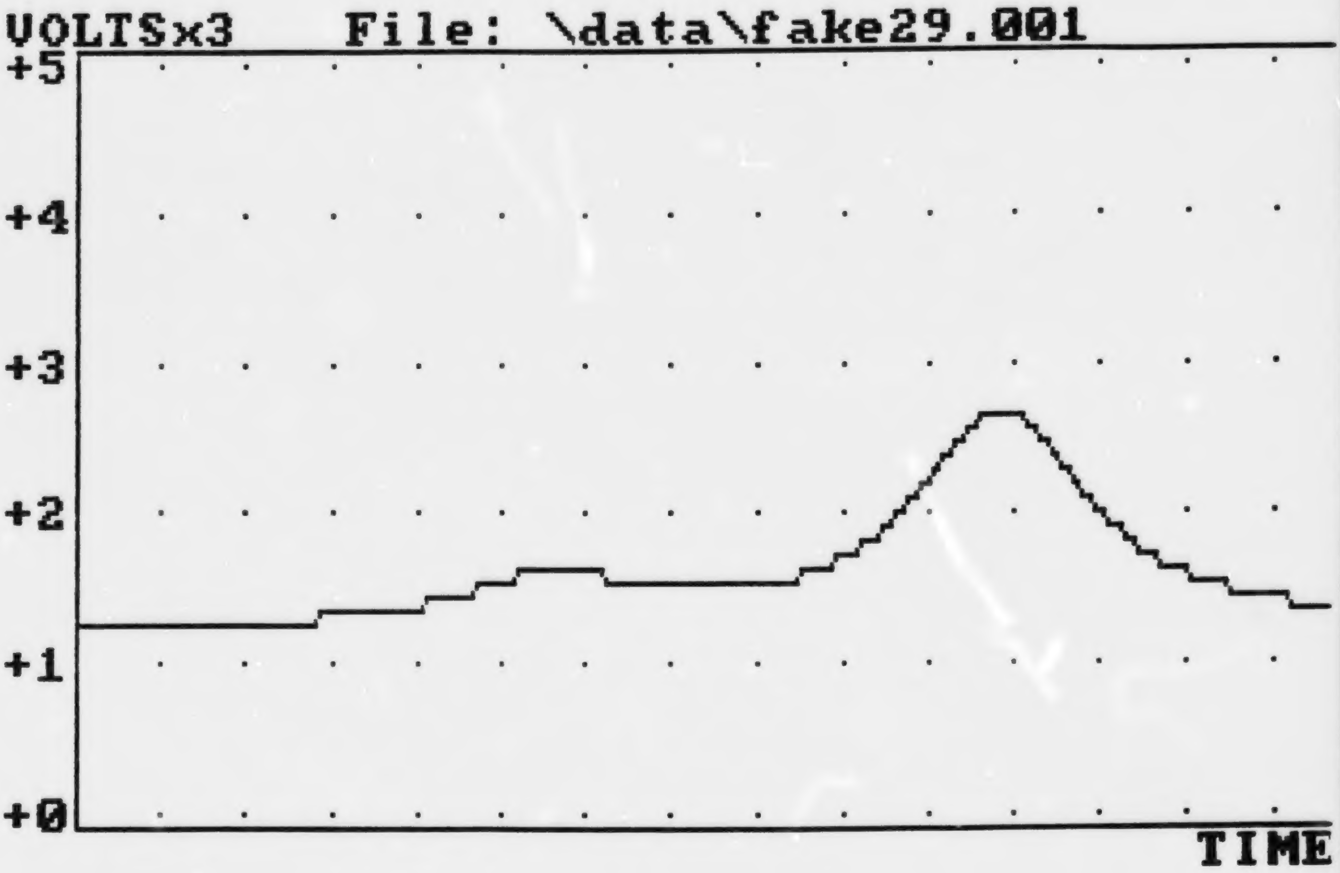
```
What's the file name ?
File: fake27.001
There are  379 data points.
 ON THE  1 ITERATION CHISQR =     266.730  NORMALIZED  RESIDUAL =    8.831
 ON THE  2 ITERATION CHISQR =     175.766  NORMALIZED  RESIDUAL =    7.873
 ON THE  3 ITERATION CHISQR =     106.061  NORMALIZED  RESIDUAL =    6.290
 ON THE  4 ITERATION CHISQR =      31.204  NORMALIZED  RESIDUAL =    3.965
 ON THE  5 ITERATION CHISQR =       2.260  NORMALIZED  RESIDUAL =     .958
 ON THE  6 ITERATION CHISQR =        .128  NORMALIZED  RESIDUAL =     .290
 ON THE  7 ITERATION CHISQR =        .127  NORMALIZED  RESIDUAL =     .288
 ON THE  8 ITERATION CHISQR =        .127  NORMALIZED  RESIDUAL =     .288
                 INITIAL                      FINAL
  MEAN1          111.000                   109.992  +/-     .02
  AMPL1        11033.963                  2985.242  +/-    6.95
  WIDTH1         113.000                    24.879  +/-     .07
  MEAN2          215.000                   214.973  +/-     .01
  AMPL2        61383.863                 69932.453  +/-   33.11
  WIDTH2         113.000                   119.976  +/-     .05
  BASELINE       376.000                   349.103  +/-     .06
  GAIN              .047                      .001  +/-     .00
 THE NUMBER OF ITERATIONS =  8
  AREA1 =    2834.     AREA2 =   56079.       TOTAL AREA =   58913.
AREA1 is  .048 of the total area.
Peak1:  .007 %         Peak2:  .140 %
```

Figure IV-48



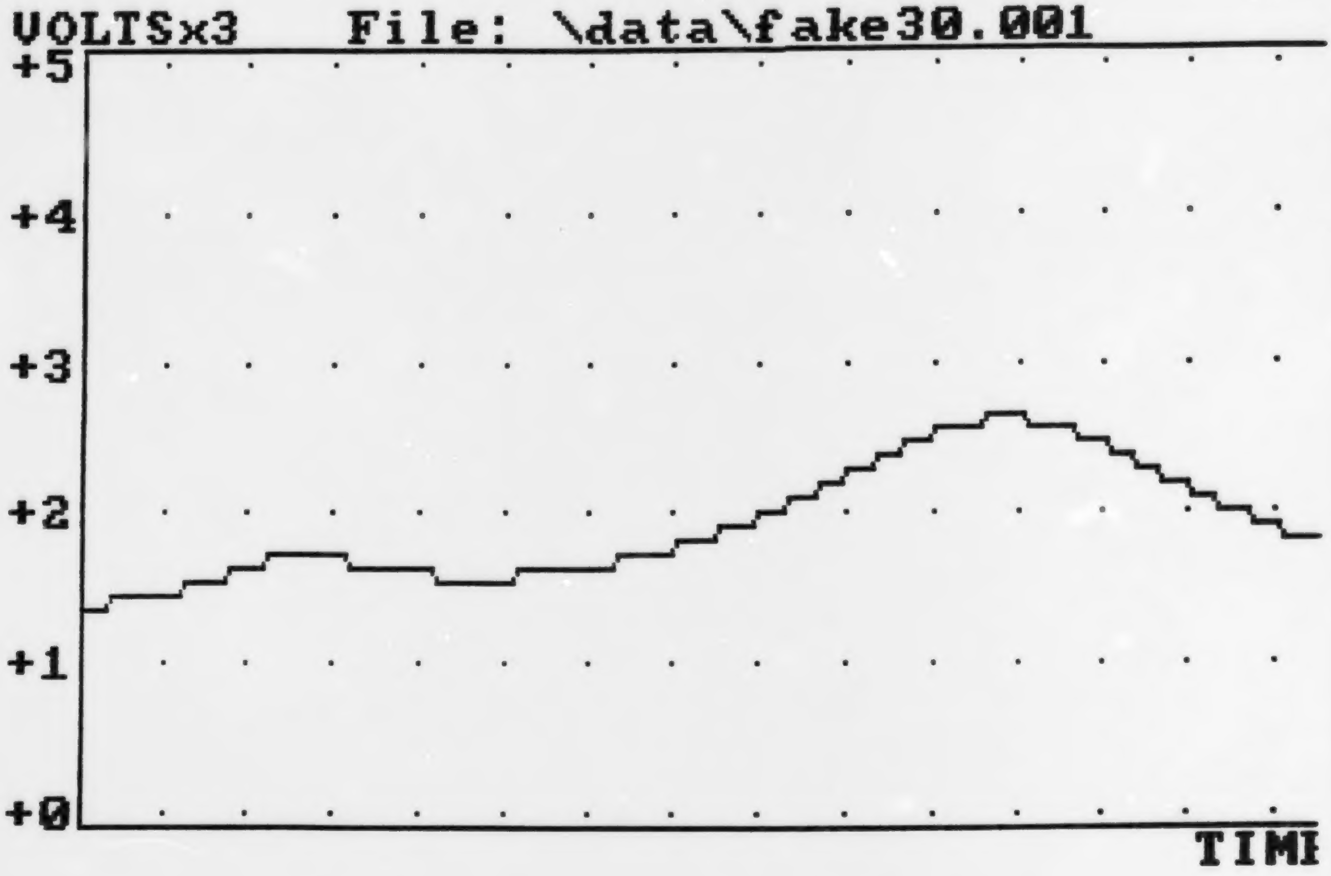
```
What's the file name ?
File: fake28.001
There are  391 data points.
 ON THE  1 ITERATION CHISQR =   1449.441  NORMALIZED  RESIDUAL =   33.625
 ON THE  2 ITERATION CHISQR =    737.806  NORMALIZED  RESIDUAL =   20.222
 ON THE  3 ITERATION CHISQR =    526.067  NORMALIZED  RESIDUAL =   15.492
 ON THE  4 ITERATION CHISQR =    358.454  NORMALIZED  RESIDUAL =   11.674
 ON THE  5 ITERATION CHISQR =    253.419  NORMALIZED  RESIDUAL =   11.752
 ON THE  6 ITERATION CHISQR =     17.275  NORMALIZED  RESIDUAL =    3.574
 ON THE  7 ITERATION CHISQR =      1.570  NORMALIZED  RESIDUAL =    1.045
 ON THE  8 ITERATION CHISQR =       .177  NORMALIZED  RESIDUAL =     .337
 ON THE  9 ITERATION CHISQR =       .166  NORMALIZED  RESIDUAL =     .330
 ON THE 10 ITERATION CHISQR =       .166  NORMALIZED  RESIDUAL =     .330
                  INITIAL                    FINAL
  MEAN1           113.000               110.014  +/-     .03
  AMPL1         12260.908              6981.645  +/-   25.22
  WIDTH1          198.000                50.071  +/-     .14
  MEAN2           213.000               215.022  +/-     .02
  AMPL2        114085.625            140326.844  +/-  139.98
  WIDTH2          198.000               200.221  +/-     .14
  BASELINE        433.000               348.741  +/-     .14
  GAIN               .066                 -.001  +/-     .00
 THE NUMBER OF ITERATIONS = 10
  AREA1 =     6285.      AREA2 =   97775.       TOTAL AREA =  104060.
AREA1 is  .060 of the total area.
Peak1:  .016 %          Peak2:  .244 %
```

Figure IV-49



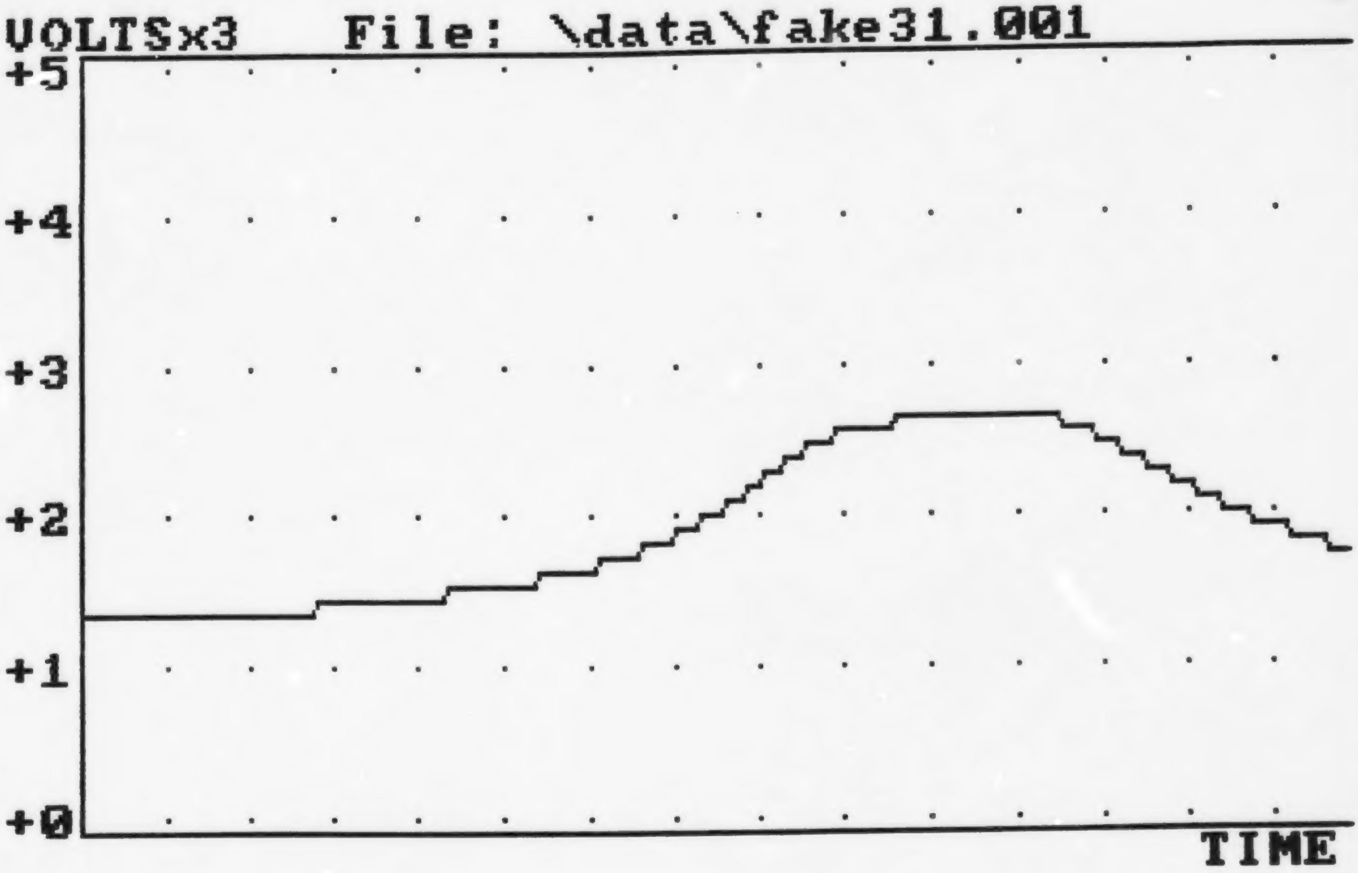
```
What's the file name ?
fake29.001
File: fake29.001
There are  353 data points.
 ON THE  1 ITERATION CHISQR =        .199  NORMALIZED  RESIDUAL =      .363
 ON THE  2 ITERATION CHISQR =        .165  NORMALIZED  RESIDUAL =      .331
 ON THE  3 ITERATION CHISQR =        .165  NORMALIZED  RESIDUAL =      .331
 ON THE  4 ITERATION CHISQR =        .165  NORMALIZED  RESIDUAL =      .331
                INITIAL                     FINAL
  MEAN1         111.000               110.006  +/-     .03
  AMPL1        6238.465              7005.810  +/-   14.54
  WIDTH1         50.000                50.164  +/-     .11
  MEAN2         215.000               214.998  +/-     .01
  AMPL2       29516.002             29977.915  +/-   12.76
  WIDTH2         50.000                49.961  +/-     .02
  BASELINE      359.000               348.961  +/-     .08
  GAIN             .008                  .001  +/-     .00
 THE NUMBER OF ITERATIONS =  4
  AREA1 =    6275.      AREA2 =   27169.      TOTAL AREA =   33444.
AREA1 is  .188 of the total area.
Peak1:  .016 %         Peak2:  .068 %


C:\SULF>
```

Figure IV-50



```
What´s the file name ?
File: fake30.001
There are  414 data points.
 ON THE  1 ITERATION CHISQR =    186.674  NORMALIZED  RESIDUAL =     8.533
 ON THE  2 ITERATION CHISQR =    173.271  NORMALIZED  RESIDUAL =     7.620
 ON THE  3 ITERATION CHISQR =     64.963  NORMALIZED  RESIDUAL =     4.987
 ON THE  4 ITERATION CHISQR =     13.594  NORMALIZED  RESIDUAL =     2.282
 ON THE  5 ITERATION CHISQR =      3.694  NORMALIZED  RESIDUAL =     1.199
 ON THE  6 ITERATION CHISQR =       .259  NORMALIZED  RESIDUAL =      .395
 ON THE  7 ITERATION CHISQR =       .165  NORMALIZED  RESIDUAL =      .333
 ON THE  8 ITERATION CHISQR =       .165  NORMALIZED  RESIDUAL =      .333
                     INITIAL                   FINAL
  MEAN1               51.000                 49.968  +/-     .03
  AMPL1            11480.249               7062.889  +/-   34.25
  WIDTH1             111.000                 50.436  +/-     .18
  MEAN2              215.000                214.999  +/-     .01
  AMPL2            59600.344              70109.491  +/-   50.49
  WIDTH2             111.000                120.090  +/-     .06
  BASELINE           381.000                348.555  +/-     .25
  GAIN                 -.029                   .001  +/-     .00
 THE NUMBER OF ITERATIONS =  8
  AREA1 =     5848.      AREA2 =   57494.       TOTAL AREA =   63342.
AREA1 is  .092 of the total area.
Peak1:  .015 %         Peak2:  .144 %
```

Figure IV-51



```
Press <ENTER> to stop or <ESC> to exit.
What´s the file name ?
File: fake31.001
There are  394 data points.
 ON THE  1 ITERATION CHISQR =     148.687  NORMALIZED  RESIDUAL =   10.363
 ON THE  2 ITERATION CHISQR =     121.949  NORMALIZED  RESIDUAL =    6.592
 ON THE  3 ITERATION CHISQR =      83.677  NORMALIZED  RESIDUAL =    5.097
 ON THE  4 ITERATION CHISQR =      31.853  NORMALIZED  RESIDUAL =    3.595
 ON THE  5 ITERATION CHISQR =       1.915  NORMALIZED  RESIDUAL =     .772
 ON THE  6 ITERATION CHISQR =        .463  NORMALIZED  RESIDUAL =     .494
 ON THE  7 ITERATION CHISQR =        .165  NORMALIZED  RESIDUAL =     .335
 ON THE  8 ITERATION CHISQR =        .165  NORMALIZED  RESIDUAL =     .335
                INITIAL                     FINAL
  MEAN1         171.000                   170.137  +/-    .05
  AMPL1        8273.034                  7183.797  +/-  76.55
  WIDTH1        122.000                    50.623  +/-    .27
  MEAN2         211.000                   215.149  +/-    .06
  AMPL2       70103.641                 69669.333  +/- 132.53
  WIDTH2        122.000                   119.787  +/-    .10
  BASELINE      377.000                   349.243  +/-    .10
  GAIN             .028                      .000  +/-    .00
 THE NUMBER OF ITERATIONS =  8
  AREA1 =    6588.     AREA2 =   56487.      TOTAL AREA =   63076.
AREA1 is  .104 of the total area.
Peak1:  .016 %        Peak2:  .141 %
```

TABLE IV-11

TRUE PARAMETERS FOR SYNTHETIC SPECTROCHRONOGRAMS

ile	Mean1	Ampl1	Width1	Mean2	Ampl2	Width2	Mean3	Ampl3	Width3	Baseline
AKE32	110	7000	50	215	70,000	120	170	10,000	50	350
AKE33	120	7000	50	215	70,000	120	180	10,000	50	350
AKE34	90	7000	50	215	60,000	120	110	10,000	50	350

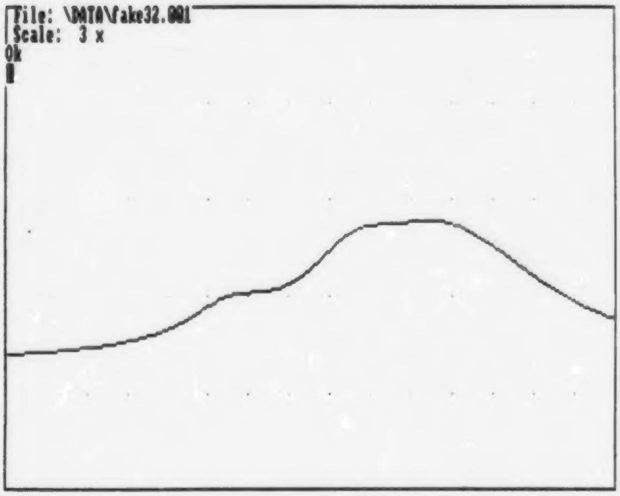


Figure IV-52

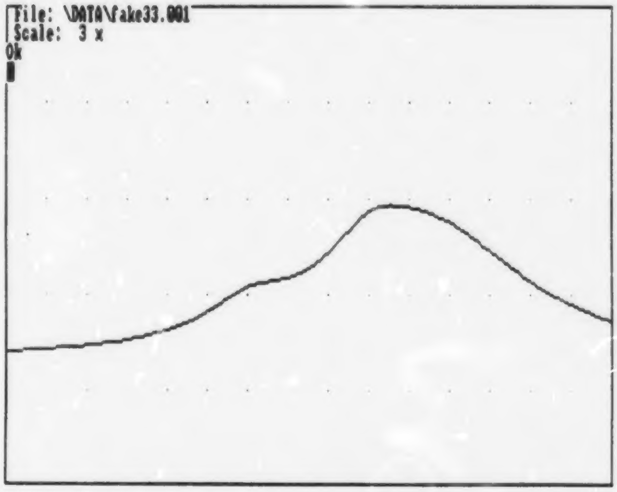


Figure IV-53

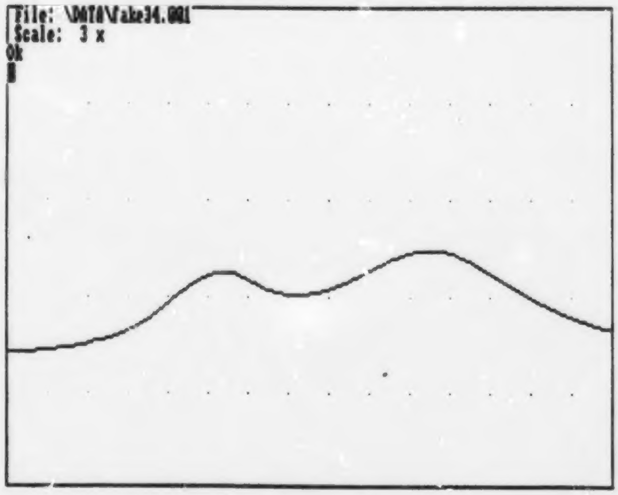


Figure IV-54

```
What's the file name ?
File: fake32.001
There are 424 data points.
What is mean1?
What is mean2?
What is mean3?
               M1     A1     W1     M2     A2     W2     M3     A3     W3
 0     0.     110  12031     70    215  98457    156    170   5000     40      354 -.007
 1    73.     108   2881     44    206  83235    132    168   4437     35      344  .002
 2    23.     112   7697     62    215  69975    121    169   9147     53      349 -.004
 3     1.     108   6576     46    215  69837    119    169  10415     50      348 -.001
 4     0.     109   7020     50    214  70120    120    170   9971     49      348  .000
 5     0.     110   7059     50    214  70184    120    170   9886     49      348  .000
 6     0.     110   7059     50    214  70184    120    170   9886     49      348  .000
                    INITIAL                    FINAL
  MEAN1             110.000                  110.068  +/-      .04 %
  AMPL1           12031.636                 7059.440  +/-      .38 %
  WIDTH1             70.000                   50.449  +/-      .32 %
  MEAN2             215.000                  214.964  +/-      .03 %
  AMPL2           98457.844                70184.530  +/-      .22 %
  WIDTH2            156.000                  120.113  +/-      .11 %
  MEAN3             170.000                  170.043  +/-      .02 %
  AMPL3            5000.000                 9886.585  +/-      .93 %
  WIDTH3             40.000                   49.583  +/-      .48 %
  BASELINE          354.000                  348.299  +/-      .04 %
  GAIN                -.007                     .000  +/--368.34 %
 THE NUMBER OF ITERATIONS =  6
  AREA1 =        6371.10      AREA2 =      57848.80    AREA3 =        9124.01
Peak1:  .016 %          Peak2:  .145 %         Peak3:  .023 %
```

Figure IV-55   Output of the DECON program for a synthetic chronogram

```
What's the file name ?
File: fake32.001
There are 424 data points.
What is mean1?
What is mean2?
What is mean3?
                  M1     A1     W1     M2     A2     W2     M3     A3     W3
 0     0.        100   2454     70    200  98457    156    150   5000     40      354 -.007
 1   278.        163  10409     98    213  73932    124    168   5175     63      345 -.007
 2   143.        136  12103    138    211  74198    127    180   5824     54      345 -.009
 3    48.        116  12149     95    213  74381    125    172   5716     49      341  .004
 4    16.        109   7856     56    213  74348    121    169   6559     33      343  .010
 5     1.        110   7257     51    213  73075    121    169   7734     42      346  .001
 6     0.        110   7115     50    214  70683    120    169   9459     48      348  .000
 7     0.        110   7059     50    214  70190    120    170   9881     49      348  .000
 8     0.        110   7059     50    214  70184    120    170   9886     49      348  .000
                      INITIAL                     FINAL
  MEAN1               100.000                   110.068   +/-      .04 %
  AMPL1              2454.006                  7059.440   +/-      .38 %
  WIDTH1               70.000                    50.449   +/-      .32 %
  MEAN2               200.000                   214.964   +/-      .03 %
  AMPL2             98457.844                 70184.541   +/-      .22 %
  WIDTH2              156.000                   120.113   +/-      .11 %
  MEAN3               150.000                   170.043   +/-      .02 %
  AMPL3              5000.000                  9886.577   +/-      .93 %
  WIDTH3               40.000                    49.583   +/-      .48 %
  BASELINE            354.000                   348.299   +/-      .04 %
  GAIN                  -.007                      .000   +/--368.33 %
 THE NUMBER OF ITERATIONS =  8
  AREA1 =        6371.10       AREA2 =      57848.81    AREA3 =        9124.00
Peak1:  .016 %           Peak2:  .145 %           Peak3:  .023 %
```

Figure IV-56   Output of the DECON program for a synthetic chronogram

```
What´s the file name ?
File: fake33.001
There are 353 data points.
What is mean1?
What is mean2?
What is mean3?
               M1     A1   W1    M2     A2   W2    M3     A3   W3
  0    0.     130   1901   70   200  95113  138   170   5000   40    358  .074
  1  311.      16   -517 -122   203  84924  128   177   7691   56    343 -.012
  2  226.    -438   1407 1605   197  92402  143   191   4877   66    355 -.077
  3  183.      62    533  -67   201  92098  137   182   5289   52    344 -.051
  4  146.     177    366  343   197  99259  145   195   1984   42    338 -.036
  5  125.     234    428   45   197  99446  144   194   2088   44    337 -.035
  6  114.     120    443   40   197  99479  144   194   2126   43    337 -.035
  7   99.     116    485   25   197  99498  144   194   2156   42    336 -.034
  8   89.     116    531   19   197  99507  143   194   2180   41    336 -.034
  9   66.     116    764    8   198  99439  143   192   2266   40    335 -.034
 10   34.     116   1211   16   199  98915  141   185   2394   39    332 -.029
 11   13.     117   2510   33   201  94441  134   180   1666   21    334 -.017
 12    6.     119   4482   44   203  85418  127   179   2987   33    340 -.004
 13    2.     119   5400   44   205  80888  124   179   4767   39    343 -.002
 14    1.     119   5870   46   206  78146  123   179   5894   41    344 -.001
 15    0.     119   6192   47   207  76254  122   179   6725   43    345 -.001
 16    0.     119   6413   48   207  74851  121   179   7372   44    346 -.000
 17    0.     119   6574   48   208  73770  121   179   7888   45    346  .000
 18    0.     119   6697   49   208  72918  121   179   8304   46    347  .000
 19    0.     120   7045   50   209  70417  119   179   9554   49    348  .001
 20    0.     120   7165   50   210  69293  119   179  10174   49    348  .002
 21    0.     120   7168   50   210  69233  119   179  10212   50    348  .002
 22    0.     120   7168   50   210  69233  119   179  10212   50    348  .002
                 INITIAL                FINAL
  MEAN1          130.000              120.113  +/-    .04 %
  AMPL1         1901.084             7168.443  +/-    .85 %
  WIDTH1          70.000               50.755  +/-    .46 %
  MEAN2          200.000              210.305  +/-    .10 %
  AMPL2        95113.742            69233.279  +/-    .66 %
  WIDTH2         138.000              119.299  +/-    .20 %
  MEAN3          170.000              179.942  +/-    .03 %
  AMPL3         5000.000            10212.493  +/-   2.43 %
  WIDTH3          40.000               50.008  +/-    .89 %
  BASELINE       358.000              348.813  +/-    .06 %
  GAIN              .074                 .002  +/-  28.64 %
 THE NUMBER OF ITERATIONS = 22
  AREA1 =     6444.27      AREA2 =      54430.55    AREA3 =      9297.29
Peak1: .016 %          Peak2: .136 %         Peak3:   .023 %
```

Figure IV-57   Output of the DECON program for a synthetic chronogram

```
What´s the file name ?
File: fake34.001
There are 353 data points.
What is mean1?
What is mean2?
What is mean3?
            M1    A1    W1    M2    A2    W2    M3    A3    W3
 0    0.    90  27125   70   210  94561  190   110   5000   40    361  .037
 1  918.    93  14989   64   209  38361   95   114   3060   32    365  .027
 2   34.    95  12462   57   209  57290  121   113   4989   45    349  .014
 3   13.    89   5476   51   209  60165  120   107  11295   55    348 -.001
 4    0.    89   7288   50   209  60090  120   110   9570   48    348 -.001
 5    0.    89   6620   49   210  60015  120   109  10376   50    348 -.001
 6    0.    89   6555   49   210  60014  120   109  10446   50    348 -.001
 7    0.    89   6557   49   210  60014  120   109  10445   50    348 -.001
                 INITIAL                    FINAL
  MEAN1           90.000                   89.452  +/-    .38 %
  AMPL1        27125.738                 6557.007  +/-   4.29 %
  WIDTH1          70.000                   49.561  +/-   1.02 %
  MEAN2          210.000                  210.013  +/-    .01 %
  AMPL2        94561.102                60014.961  +/-    .13 %
  WIDTH2         190.000                  120.021  +/-    .10 %
  MEAN3          110.000                  109.605  +/-    .21 %
  AMPL3         5000.000                10445.733  +/-   2.74 %
  WIDTH3          40.000                   50.366  +/-    .69 %
  BASELINE       361.000                  348.570  +/-    .05 %
  GAIN              .037                    -.001  +/- -91.51 %
 THE NUMBER OF ITERATIONS =  7
  AREA1 =      5794.64      AREA2 =     47118.71    AREA3 =       9349.40
Peak1:  .014 %          Peak2:  .118 %          Peak3:  .023 %
```

Figure IV-58   Output of the DECON program for a synthetic chronogram

```
What's the file name ?
File: fake34.001
There are 353 data points.
What is mean1?
What is mean2?
What is mean3?
              M1     A1      W1    M2     A2     W2     M3     A3     W3
 0     0.     80   14312    70    200   93144   192    120    5000    40       361  .037
 1  1541.     95   18877    88    220   34436    88    110    6804    51       363 -.008
 2   178.    109   19878    72    211   53685   119    105    2254    52       347  .009
 3    56.     96   16278    60    209   61558   121    132    1679    48       341  .014
 4    11.     99   16431    60    210   58635   118    113    2963    86       343  .015
 5     3.    100   16386    60    210   58683   118    109    2859    64       343  .016
 6     2.    100   16338    61    210   58693   118    110    2819    57       343  .016
 7     2.    100   16298    61    210   58728   118    111    2783    52       343  .016
 8     2.     99   16247    61    210   58798   118    112    2739    48       343  .017
 9     1.     98   15860    60    210   59566   119    117    2461    37       344  .013
10     1.     97   14246    58    209   60292   120    114    3500    41       345  .007
11     1.     96   13174    57    209   60431   120    114    4353    42       345  .006
12     1.     95   12190    56    209   60441   120    113    5193    43       346  .004
13     0.     94   11330    55    209   60424   120    113    5942    44       346  .003
14     0.     93   10578    54    209   60396   120    112    6613    45       347  .002
15     0.     93    9924    54    209   60360   120    112    7206    46       347  .002
16     0.     92    9360    53    209   60322   120    111    7725    47       347  .001
17     0.     92    8879    52    209   60283   120    111    8175    47       347  .001
18     0.     91    8471    52    209   60246   120    111    8562    48       347  .000
19     0.     90    7097    50    209   60095   120    110    9896    49       348 -.000
20     0.     89    6544    49    210   60016   120    109   10455    50       348 -.001
21     0.     89    6556    49    210   60014   120    109   10445    50       348 -.001
22     0.     89    6556    49    210   60014   120    109   10445    50       348 -.001
                   INITIAL                       FINAL
  MEAN1             80.000                      89.452  +/-     .38 %
  AMPL1          14312.219                    6556.991  +/-    4.29 %
  WIDTH1            70.000                      49.561  +/-    1.02 %
  MEAN2            200.000                     210.013  +/-     .01 %
  AMPL2          93144.961                   60014.959  +/-     .13 %
  WIDTH2           192.000                     120.021  +/-     .10 %
  MEAN3            120.000                     109.605  +/-     .21 %
  AMPL3           5000.000                   10445.750  +/-    2.74 %
  WIDTH3            40.000                      50.366  +/-     .69 %
  BASELINE         361.000                     348.570  +/-     .05 %
  GAIN                .037                       -.001  +/-  -91.50 %
 THE NUMBER OF ITERATIONS = 22
  AREA1 =        5794.62      AREA2 =      47118.71     AREA3 =         9343.41
Peak1:  .014 %          Peak2:  .118 %           Peak3:  .023 %
```

Figure IV-59   Output of the DECON program for a synthetic chronogram

Another set of synthetic spectrochronograms with two peaks was generated to test the deconvolution algorithm. The true values for these spectrochronograms are given in Table IV-12. Table IV-13 compares the area found by the deconvolution algorithm with the known true values for these synthetic spectrochronograms. For file fake43.001, which has a peak1 much smaller than peak2 (0.78% of the peak area) the deconvolution algorithm failed to correctly locate peak1. The deconvolution algorithm's final estimate of the position of peak1 was at 237 rather than the correct value of 170. The ability of the algorithm to correctly identify the minor peak depends also on how closely it is overlapped with the major peak.

The deconvolution algorithm is based on Marquardt's algorithm. Marquardt's algorithm is an interpolation between the gradient search method and the method of linearizing the fitting function. The parameter lambda defines the proportion of gradient search included. Lambda is varied by the algorithm in the course of finding the best fit. However, the initial value of lambda is written into the program and may influence the success of the algorithm. The file fake37.001 was selected as moderately difficult for the deconvolution algorithm. Using this one synthetic spectrochronogram, various initial values of lambda were used. The results of these tests are given in Table IV-14. Although an initial lambda of 0.0007 has the smallest error for AREA1, it is wrong. It found MEAN1 at 2519 instead of 170. For this test, the final value of chi squared should be taken as the criterion for success. Based on this, a value of 0.0005 or 0.0006 is the best choice for the initial value of lambda.

TABLE IV-12

SYNTHETIC SPECTROCHRONOGRAMS HAVING 2 PEAKS

Filename	Mean1	Ampl1	Width1	Mean2	Ampl2	Width2	Baseline	Area1	Area2
fake35.001	170	3000	50	215	70,000	120	350	2756	56872
fake36.001	170	7000	50	215	70,000	120	350	6424	56609
fake37.001	170	1000	50	215	70,000	120	350	920	57049
fake43.001	170	500	50	215	70,000	120	350	465	59318

TABLE IV-13

TRUE AND FOUND AREAS FOR SPECTROCHRONOGRAMS

Filename	True Area1	Area1 Found	% Error	True Area2	Area2 Found	% Error
fake35.001	2756	2905	+5.4	56872	56733	-0.2
fake36.001	6424	6581	+2.4	56609	56383	-0.4
fake37.001	920	1196	+30.0	57049	56583	-0.8
fake43.001	465	2485	+434.4	59318	57370	-3.3

TABLE IV-14

VARYING INITIAL VALUES OF LAMBDA FOR MARQUARDT'S ALGORITHM*

Initial lambda	No. of Iterations	Final Chisqr	Area1 Found	% Error	Area2 Found	% Error
0.0010	18	0.751	6804	+640	62130	+8.9
0.0009	15	0.751	6804	+640	62130	+8.9
0.0008	25	0.751	6823	+642	62132	+8.9
0.0007	11	1.423	1097	+19.2	58814	+3.1
0.0006	11	0.141	1196	+30.0	56583	-0.8
0.0005	11	0.141	1196	+30.0	56583	-0.8
0.0004	61	0.858	18046	+1861	60988	+6.9
0.0003	27	0.749	6989	+659	62153	+9.0
0.0002	26	0.749	6989	+659	62153	+9.0
0.0001	18	0.751	6804	+640	62130	+8.9

* Using File fake37.001

IV.9 COMPENSATION FOR DETECTOR NON-LINEARITY

The response of the infrared detector to SO_2 is not linear. A plot of the peak height versus amount of sulfur does not form a straight line. In the Leco instrument there is software to allow the operator to linearize the instrument's response. In this project, the Compaq computer takes the raw signal from the SO_2 detector as input. Therefore, there was a need for software on the Compaq to accomplish the same linearization process.

To construct the calibration curve (instrument response versus amount of sulfur) chronograms were acquired for benzene sulfonic acid (BSA) at 550°C. BSA was selected because its chronogram has a single narrow peak at 550°. A typical BSA chronogram is shown in Figure IV-60. For each chronogram the peak height was measured with respect to the baseline (the average of the first 40 points in the chronogram) by using the program AREA. Table IV-15 lists the chronograms used to construct the calibration curve. The calibration curve is made up of 15 straight line segments, each resulting from peak heights applicable in that region. Each of the 15 straight lines can be represented as:

$$Y_2 = m_2X + b_2$$

where m_2 is its slope and b_2 is its intercept. There is a single straight line with slope m_1 and intercept b_1 which covers the same range of sulfur weights as the 15 segments. (See Figure IV-61).

In order to effect the linearization any point on one of the 15 segments must be converted to the corresponding point on the single straight line, i.e., at the same abscissa value. The ordinate value, y_1, on the single line is calculated from the ordinate value, Y_2, on one of the 15 segments by the equation:

$$Y_1 = m_1/m_2 * (Y_2 - b_2) + b_1$$

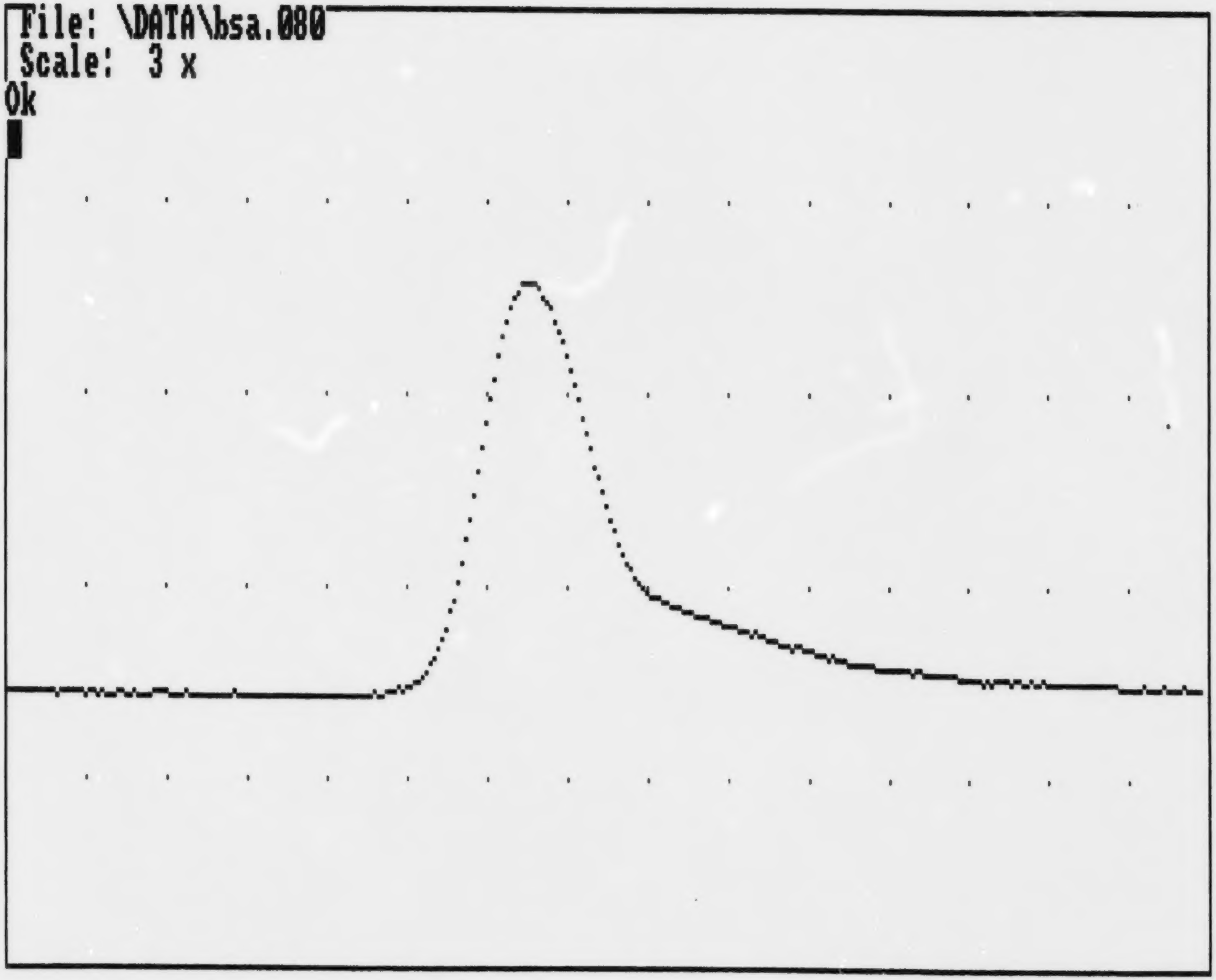


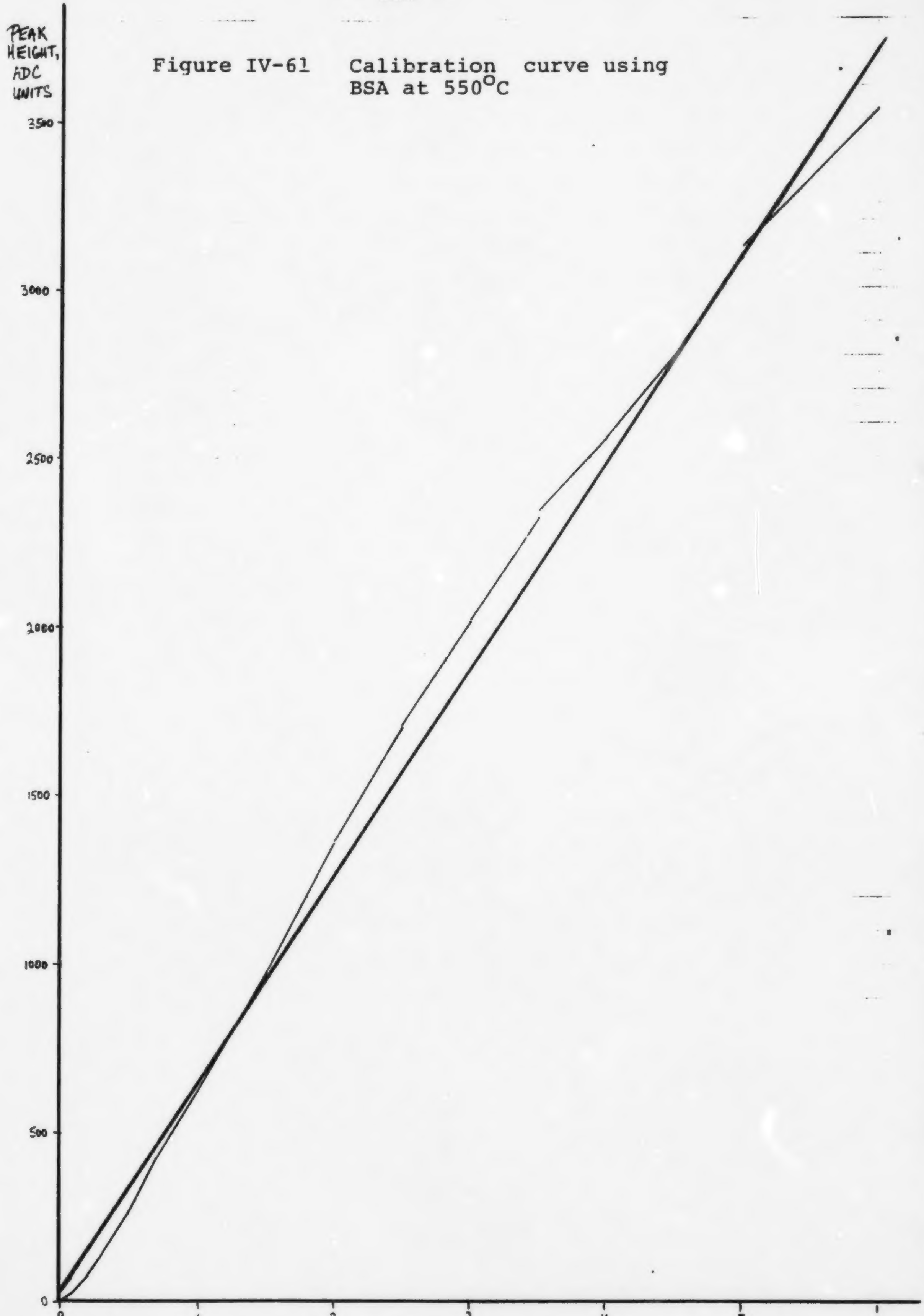
Figure IV-60 A typical spectrochronogram for benzenesulfonic acid at 550^{o}C.

TABLE IV-15

BSA CHRONOGRAMS USED IN CALIBRATION CURVE

FILE	mg S	PEAK HEIGHT
BSA .060	0.11	35.3
BSA .061	0.11	34.3
BSA .062	0.10	28.8
BSA .063	0.20	76.3
BSA .064	0.20	75.4
BSA .065	0.20	84.6
BSA .066	0.30	143.4
BSA .067	0.30	132.9
BSA .068	0.30	141.6
BSA .069	0.50	267.5
BSA .070	0.50	290.1
BSA .071	0.50	256.4
BSA .072	0.71	436.9
BSA .073	0.71	426.7
BSA .074	0.70	435.3
BSA .075	1.00	684.1
BSA .076	0.99	601.3
BSA .077	1.02	630.9
BSA .078	1.01	585.1
BSA .079	1.01	673.1
BSA .080	1.01	583.7
BSA .081	1.00	637.5
BSA .028	1.54	1061.1
BSA .029	1.54	997.8
BSA .030	1.52	920.8
BSA .031	1.54	941.5
BSA .032	1.53	1089.8
BSA .033	2.00	1366.9
BSA .034	2.06	1359.2
BSA .035	2.06	1446.3
BSA .036	2.52	1717.6
BSA .037	2.58	1720.1
BSA .038	2.55	1757.2
BSA .039	3.09	2077.6
BSA .040	3.03	2071.3
BSA .041	3.03	1999.8
BSA .042	3.54	2410.2
BSA .043	3.57	2355.6
BSA .044	3.64	2363.6
BSA .045	4.08	2641.7
BSA .046	4.08	2474.4
BSA .047	4.08	2632.8
BSA .048	4.04	2612.9
BSA .049	4.59	2872.0
BSA .050	4.59	2863.1
BSA .051	4.68	2839.2
BSA .052	5.05	3115.5
BSA .053	5.05	3208.6
BSA .054	5.20	3188.4
BSA .055	6.12	3571.8
BSA .001	6.18	3748.1
BSA .002	5.94	3435.1
BSA .003	6.18	3514.3
BSA .004	6.12	3591.1

Figure IV-61 Calibration curve using BSA at 550°C



This is the function of the FORTRAN program, LINEAR. It reads every point in the input chronogram, finds the point on one of the 15 segments and calculates the corresponding point on the single straight line. Thus, a new, linearized chronogram is generated. This has the effect of lowering some parts of the curve and raising others. The source code file, LINEAR.FOR, contains the values for m_1 and b_1 and m_2 and b_2 for the 15 segments.

V. DISCUSSIONS AND CONCLUSIONS

The three classes of sulfur compounds: organic, pyritic and sulfate are always present in coal but their content can vary considerably from one coal to another. In this study an analysis method has been developed which provides a complete separation of sulfatic sulfur. However, only partial separation between organic and pyritic sulfur was possible.

The major forms of organic sulfur identified in coal are: thiophenes, aromatic and aliphatic sulfides, thiols, and thiophenols. These compounds are usually uniformly distributed in the organic matrix. From the temperature programmed oxidation study maxima of SO_2 from the organic sulfur compounds occurred at about 320°C (attributed to non-aromatic coal structures) and at about 480° (attributed to aromatic structures) with the pyritic SO_2 peak at about 430°C.

In our study partial speciation between organic and pyritic sulfur has been attained for some coals with good spectro-chronogram reproducibility for all. Application of a peak deconvolution method requires knowledge of a relationship between the major organic sulfur groups present in coal. From studies reported in the literature (Karr, 1979) and in our study the spectro-chronograms appear very much different for some coals. The differences appear to be primarily due to different distribution within the organic sulfur functional groups.

In addition to the above variations most of the desulfurization processes that remove the pyritic sulfur production very inconsistent effects on the organic sulfur. The reason is that even if the total organic sulfur in different coals is the same, its distribution among different organic sulfur groups might be different.

Since the rate of reaction of each group is different one can expect that different processes will have different effects on the individual sulfur groups.

The quantitative identification of the sulfur forms through the deconvolution methods relies on the k-value which is the ratio of the organic sulfur in peak1 to pyritic sulfur in peak2. The validity of this assumption has to be re-evaluated.

In addition to the above, the coal samples taken directly from the coal deposits should be investigated. The coal sample preparation has to be also considered. The present study has been based solely on the Leco coals. In the coals received from Argonne Laboratory no first peak, (attributed to a non-aromatic structure) has been observed in most of their coal samples and the present deconvolution technique would not be applicable to these coal samples.

In fresh coals, sulfates rarely exceed 0.3%. The exposure of coal to air and moisture increases sulfate content due to the pyrite oxidation to iron sulfate. It is thus not clear whether some of the Leco coals, which contained considerable amounts of sulfates, were modified during the sample preparation process or whether it was natural decomposition of the coal samples. It is thus strongly recommended that the present method of analyses be applied to coal samples that have been prepared in dry and non-oxidizing media like nitrogen or argon and compared with coal samples from the same region but prepared in air.

The attempts to resolve the observed chronograms into three peaks and get their areas accurately was seen to work in some cases but not in others. It was inconsistent. The deconvolution program works iteratively and with every iteration it attempts to maximize the goodness of fit. When the goodness of fit improves by moving a parameter in a certain direction, the program keeps moving the parameter in that direction. However, the situation is complicated by the fact that eleven parameters were being optimized at once. Sometimes meaningless results were found (such as negative values for peak height or width). This was because Marquardt's algorithm is a purely mathematical operation and does not take into account which values are physically meaningless and should be avoided. This problem was reduced by introducing constraints on the values that the parameters can assume in the program outside of Marqhardt's algorithm.

The studies using synthetic spectrochronograms showed that the deconvolution algorithm could correctly find all the parameters (position, width, and amplitude) of three peaks even when they are severely overlapped. This is not surprising because the synthetic chronograms are exactly made up of Lorentzian curves. The real chronograms on the other hand are not exactly Lorentzian curves exactly and have contributions from experimental uncertainties. These uncertainties appear most often as slight lumps and bumps superimposed on the normal chronogram. The approximate appearance of replicate chronograms appears to be the same with the major features reproducible. The conclusion of the Phase II studies that the chronograms were reproducible was in the sense that peak positions and heights as measured on a chart recorder were precise to within a few percent. However, because there are changes in the shape of the peaks, it is not possible for the deconvolution program to reliably resolve peaks that are severly overlapped, as is the pyrite peak and the second organic peak. In order for deconvolution to resolve these peaks the chronograms would have to be reproducible in peak shape as well as peak height and position. Furthermore, that peak shape would have to conform to the Lorentzian model as it exists in the deconvolution program.

In the work done with the k-factor method, which requires deconvolution into only two peaks, the results appear to have better reliability and precision. This was based on the k-factor, calculated from all ultimate analysis data, and then applying that k-factor to the deconvoluted peak areas.

VI. REFERENCES

1. Karr, C., 1979, Analytical Methods for Coal and Coal Products Edited by C. Karr, Jr., Academic Press.

2. Maijgreen, B., Hubner, W., Norrgard, K. and Sundvall, S.B., 1983 Fuel, 62, 1076-78.

3. Moon, K.S., 1987 Mineral Science Laboratories Division Report, ERP/MSL 85-97.

4. Moon, K.S., and Sirois, L.L., 1985 Mineral Science Laboratories Division Report ERP/MSL 85-97.

5. Yurovskii, A.Z., 1974 Sulfur in Coals, TT70-57216, Academy of Sciences of the U.S.S.R., Institute of Mineral Fuels.

VII. APPENDIX

SOFTWARE LISTINGS

```
C
C     PROG DECON
C
C     PURPOSE
C     TO RESOLVE THE DOUBLE PEAKED CURVE FROM THE RAW SULPHUR
C     IN COAL DATA, INTO TWO SEPARATE PEAKS. THIS IS ACCOMPLISHED
C     BY PERFORMING A NON-LINEAR FIT TO THE RAW DATA.
C
C

        INTEGER * 4 COUNT

        REAL * 4 A1,A2,SO(512)  ,SF(512),TIME(512)
        REAL * 4 XYFIT(600), MINSO, AREA, AREA1, AREA2,AREA3,F(512)
        REAL * 4 TOTAREA,CALIB,ORGA,ORGA2,PYRA,ORG,PYR,TOTRES,NORMRES
        REAL * 4 BASELINE,MEAN1,WIDTH1,AMPL1,MEAN2,WIDTH2,AMPL2
        REAL * 4 GAIN, Y(512),CHIS2,CHISQR,samwt,gramorg,grampyr,pctorg
        REAL * 4 pctpyr,HEADER(33),DELTAA(10),SIGMAA(10),kfac

        REAL*8 SPDR(8)   ,R8BLANK/'        '/,V(10),OLDV(10)
        REAL*8 DERIV(512,10),Y1(512),Y2(512),IN(10)
        REAL*8 PARNAME (10),YFIT(512),calfac

      LOGICAL * 4 EOF

        CHARACTER * 18  filename

      EXTERNAL FNCTN

      DATA PARNAME/'MEAN1   ','AMPL1   ','WIDTH1  ','MEAN2   ',
     +'AMPL2   ','WIDTH2  ','BASELINE','GAIN    ','MEAN1   ','WIDTH3  '/
      DATA NTERMS/8/
      DATA NRMAX/100/,NSMAX/600/,NPK/10/
C
C       Read in the data at this point.
C
        call rdfile(NPTS,Y,filename,samwt)
C       do 80 K = 1,NPTS
C  80   write (0,100) K, Y(K)
C 100   format (I8,F10.2)

        open (9,file='LPT1',mode='WRITE')
        write (9,111) filename
  111   format (' File: ',A15)

      DO 200 I = 1,512
      SF(I) = 0.0
      XYFIT(I) = 0.0
 200  YFIT(I) = 0.0
      WRITE ( 6, 300 ) NPTS
      WRITE ( 9, 300 ) NPTS
 300  FORMAT ( ' There are ',I4,' data points.')

C
C     GET INITIAL ESTIMATES FOR THE 9 PARAMETERS
C
      CALL ESTIM (Y,NPTS,BASELINE,MEAN1,WIDTH1,AMPL1,MEAN2,WIDTH2,
     +AMPL2,GAIN)

      IN(1)= MEAN1
      IN(2)= AMPL1
      IN(3)= WIDTH1
      IN(4)= MEAN2
      IN(5)= AMPL2
      IN(6)= WIDTH2
      IN(7)= BASELINE
      IN(8)= GAIN
      V(1)=MEAN1
      V(2)=AMPL1
      V(3)=WIDTH1
      V(4)=MEAN2
      V(5)=AMPL2
      V(6)=WIDTH2
      V(7)=BASELINE
      V(8)=GAIN
      DO 400 J=1,NPTS
400   TIME(J)=J-1
      CHIS2=1.0E35
      DO 500 I=1,8
      OLDV(I)=V(I)
500   DELTAA(I)=V(I)/10
c      FLAMDA = .001
        flamda = .0005
      CHISQR=0
      COUNT=0
        write (6,600)
        write (9,600)
 600    format (14X,2HM1,4X,2HA1,4X,2HW1,4X,2HM2,4X,2HA2,4X,2HW2)
        write (6,650) count,chisqr, ( int(v(I)), I = 1,7 ),v(8)
        write (9,650) count,chisqr, ( int(v(I)), I = 1,7 ),v(8)
 650    format (I3,1X,F5.0,1X, 2(2X,I4,1X,I6,1X,I4),3X,I4,1X,F4.3)

 700  IF (ABS(CHIS2-CHISQR) .GT. .001*CHIS2 .OR. (COUNT .LT. 4)) THEN
      COUNT=COUNT+1
      CHIS2=CHISQR

c       if ( V(1) .gt. 200. ) V(1) = IN(1)
c       if ( V(2) .gt. 60000. ) V(2) = IN(2)
c       if ( V(3) .gt. 70 ) V(3) = 1.01 * IN(3)
c       if ( V(2) .gt. 50000. .or. V(3) .gt. 200. ) then
c               V(2) = IN(2)
c               V(3) = IN(3)
c       endif
c       if ( V(4) .lt. V(1) .or. V(4) .gt. 350. ) V(4) = IN(4)
c       if ( V(5) .gt. 150000. ) V(5) = IN(5)
c       if ( V(6) .gt. 200. ) V(6) = IN(6)

      CALL CURFIT(TIME,Y,NPTS,NTERMS,V,DELTAA,SIGMAA,FLAMDA
     +,YFIT,CHISQR,FNCTN,DERIV,Y1,Y2)

        do 800 J = 1,6
 800    v(J) = abs ( v(J) )

c      WRITE(6,465) COUNT,CHISQR,NORMRES
c465   FORMAT(2X,'ON THE ',I2,' ITERATION CHISQR = ',F10.3,'  NORMALIZED
c     + RESIDUAL = ',F8.3)

        write (6,1000) count,chisqr, ( int(v(I)), I = 1,7 ),v(8)
        write (9,1000) count,chisqr, ( int(v(I)), I = 1,7 ),v(8)
 1000   format (I3,1X,F5.0,1X, 2(2X,I4,1X,I6,1X,I4),3X,I4,1X,F4.3)
```

```
      DO 1100 I=1,8
1100  DELTAA(I)=(V(I)-OLDV(I))/10
      DO 1200 I=1,8
1200    OLDV(I)=V(I)
        ELSE
                GOTO 1300
        ENDIF
        GOTO 700
 1300   WRITE (6,1400)
        WRITE (9,1400)
 1400   FORMAT (18X,'INITIAL',16X,'FINAL')
      WRITE (6,1500) (PARNAME(J),IN(J),V(J),SIGMAA(J),J=1,8)
      WRITE (9,1500) (PARNAME(J),IN(J),V(J),SIGMAA(J),J=1,8)
 1500   FORMAT(2X,A9,2X,F12.3,9X,F12.3,2X,'+/-',F7.2)
      WRITE(6,1600) COUNT
      WRITE(9,1600) COUNT
 1600   FORMAT(2X,'THE NUMBER OF ITERATIONS = ',I2)
      AREA1 = 0.0
      AREA2 = 0.0
      DO 1700 II = 1,NPTS
      AREA1 = AREA1 + 0.159154943 * V(2) * V(3)/((II- V(1))**2 +
     +(V(3)/2.)**2)
 1700   AREA2 = AREA2 + 0.159154943 * V(5) * V(6)/((II- V(4))**2 +
     +(V(6)/2.)**2)
      TOTAREA = AREA1 + AREA2
      WRITE (6,1800) AREA1, AREA2, TOTAREA
      WRITE (9,1800) AREA1, AREA2, TOTAREA
 1800   FORMAT (2X,' AREA1 = ' ,F8.0,'      AREA2 = ',F8.0,
     +'       TOTAL AREA = ', F8.0)


        call wrfile(NPTS,YFIT,filename)

        WRITE  (6, 2000)
2000    format (' What''s the k factor (orgl/pyr) ?')
        read (0,2100) kfac
2100    format (f10.8)
        pyra = areal/kfac
        orga2 = area2 - pyra
        orga = areal + orga2

        gramorg = orga/40000000.
        grampyr = pyra/40000000.
        pctorg = 100. * gramorg/samwt
        pctpyr = 100. * grampyr/samwt
        write (6,2200) kfac,pctorg,pctpyr
        write (9,2200) kfac,pctorg,pctpyr
2200    format (' k factor: ',f7.4,'     Organic: ',f8.2,
     +' %        Pyritic: ',f8.2,' %')
        write (6,2300)
        write (9,2300)
2300    format ('0')

        WRITE  (6, 2400)
2400    format (' What''s the k factor (orgl/pyr) ?')
        read (0,2500) kfac
2500    format (f10.8)
        pyra = areal/kfac
        orga2 = area2 - pyra
      orga = area1 + orga2

      gramorg = orga/40000000.
      grampyr = pyra/40000000.
      pctorg = 100. * gramorg/samwt
      pctpyr = 100. * grampyr/samwt
      write (6,2600) kfac,pctorg,pctpyr
      write (9,2600) kfac,pctorg,pctpyr
2600  format (' k factor: ',f7.4,'     Organic: ',f8.2,
    +' %          Pyritic: ',f8.2,' %')
      write (6,2700)
      write (9,2700)
2700  format ('0')

      DO 2800 I=1,NPTS
     XYFIT(I) = YFIT(I)
     SF(I)=XYFIT(I)-Y(I)
2800  CONTINUE
      RETURN
8000 WRITE (6,8010)
8010 FORMAT (2X,'ERROR READING INPUT')
9000 WRITE(6,9010)
9010 FORMAT(//' MISSING OR INVALID INPUT DIRECTIVE: '    )
9100 WRITE(6,9110)
9110 FORMAT(//'CAN T READ PROPERLY')
      end
```

```
Listing of RDFILE.FOR,  11-Mar-1988  14:06:10                        Page 1

C
C      SUBROUTINE RDFILE
C
C      This subroutine reads data from a file which must be in the
C      \data directory. The file must be of the type written by COAL.C
C      or FAKE.C. This file structure is: 14 bytes of header, then
C      N times: 4 bytes of time and 2 bytes of data. (There are N data
C      points.) Only the 2 data bytes out of every 6 are extracted.
C      This subroutine prompts the user for the filename. This
C      subroutine also extracts sample weight as a 4 byte real number
C      from the header (starting at the 7th byte of the header).
C
C      Description of parameters:
C
C        N - the number of data points read from the file
C
C        Y - the array of data points (dimensioned to 512)
C
C        samwt - sample weight in grams. This is read from the
C            data file header and passed back to the calling program.

      SUBROUTINE RDFILE (N,Y,filename,samwt)

      INTEGER * 2  X(512)
      REAL * 4  Y(512), samwt
      CHARACTER * 18 filename

   30 WRITE  (*, 40)
   40 format (' What''s the file name ?')
      read (0,45) filename
   45 format (A15)
      write (*,48) filename
   48 format (' File: ',A15)
c
c     ************ read the sample weight **************
c
      open (1,file = '\data\'// filename, recl=6,
     +form='UNFORMATTED', access='DIRECT',mode = 'READ',
     +err = 9000, status = 'OLD')
      read (1,rec=2, end = 1000 ) samwt
 1000 close(1)
c
c     ************ read the actual data *****************
c
      open (1,file = '\data\'// filename, recl=2,
     +form='UNFORMATTED', access='DIRECT',mode = 'READ',
     +err = 9000, status = 'OLD')
      do 50 N = 1,512
   50 read (1,rec = 3 * N + 7, end = 60 ) X(N)
   60 close(1)
      N = N - 1
      do 100 I = 1,N
  100 Y(I) = real (X(I))
      go to 5000
 9000 write (*, 9010)
 9010 format (' RDFILE: Can''t open that file.')
      go to 30

 5000 return
      end
      SUBROUTINE WRFILE (N,YFIT,filename)
c
c     This subroutine writes a file that has N points of fitted
c     data, YFIT(I). Each real point is converted to INT*2 before
c     being written to the file. The written file has the same format
c     as the files stored by the data acquisition software. The name
c     of the written file is the filename passed to this subroutine.
c     The file is written in directory \data.
c
      INTEGER * 2  X(512)
      REAL * 8     YFIT(512)
      CHARACTER * 18 filename

      do 100 I = 1,N
  100 X(I) = INT2 (YFIT(I))
      open (2, file = '\data\f'// filename, recl = 2,
     +form = 'UNFORMATTED', access = 'DIRECT', mode = 'WRITE',
     +err = 9000)
      do 200 I = 1,N
  200 write (2, rec = 3 * I + 7) X(I)
      close (2)
      go to 5000
 9000 write (*,9010)
 9010 format (' WRFILE: Can''t open that file.')
 5000 return
      end
```

```
Listing of ESTIM.FOR,  17-Aug-1988  16:45:32                      Page 1

$NOTRUNCATE
C
C      SUBROUTINE ESTIM
C
C      THIS SUBROUTINE ESTIMATES THE PEAK PARAMETERS AND BASELINE
C      IN THE INPUT ARRAY, T.
C
       SUBROUTINE ESTIM (T,  NPTS,BASELINE,MEAN1,WIDTH1,AMPL1,MEAN2,
     +WIDTH2 ,AMPL2,GAIN)
C
       INTEGER*4 I, I1, I2, NPTS
       REAL*4 BASELINE, MAX1, MAX2, T(NPTS), PKHT1, PKHT2, HALFHT1
       REAL*4 HALFHT2, WIDTH1, WIDTH2R,WIDTH2L,AMPL1,AMPL2R,AMPL2L,P(512)
       REAL*4 WIDTH2,AMPL2,MEAN1,MEAN2,GAIN
C
       BASELINE = 1E6
       MAX2 = 0.
       DO 100 I = 1,NPTS
       IF ( T(I) .GT. MAX2 ) THEN
          MAX2 = T(I)
          MEAN2 = I
       ENDIF
 100   IF ( T(I) .LT. BASELINE ) BASELINE = T(I)
       PKHT2 = MAX2 - BASELINE
       HALFHT2 = BASELINE + PKHT2/2.
       DO 200 I = 1,MEAN2
       IF ( T(I) .GE. HALFHT2 ) THEN
          I1 = I
          GO TO 300
       ENDIF
 200   CONTINUE
 300   DO 400 I = MEAN2, NPTS
       IF ( T(I) .LE. HALFHT2 ) THEN
          I2 = I
          GO TO 500
       ENDIF
 400   CONTINUE
 500   WIDTH2  = I2 - I1
       IF ( WIDTH2 .GT. 200 ) WIDTH2 = 100
       AMPL2  = 1.57 * WIDTH2  * PKHT2
       DO 600 I = 1,NPTS
 600   P(I)   = T(I)-0.15915*AMPL2*WIDTH2/((I-MEAN2)**2.+(WIDTH2/2.)
     +**2.)
c      Do 650 I = 1,NPTS,5
c 650  write (*,660) (P(L), L=I,I+4)
c 660   format (5f10.4)
       MAX1 = 0.
       DO 700 I = 1,MEAN2-30
       IF ( P(I) .GT. MAX1 ) THEN
          MAX1 = P(I)
          MEAN1 = I
       ENDIF
 700   CONTINUE
       PKHT1 = MAX1 - BASELINE
       WIDTH1 = WIDTH2
       AMPL1 = (1.57 * WIDTH1 * PKHT1)
       GAIN = (T(NPTS) - T(1))/NPTS
       RETURN
       END
```

```
C
C     SUBROUTINE CURFIT
C
C     PURPOSE
C        MAKE A LEAST SQUARES FIT TO A NON-LINEAR FUNCTION USING
C        MARQUARDT'S ALGORITHM.
C
C     DESCRIPTION OF PARAMETERS:
C
C       X - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE.
C
C       Y - ARRAY OF DATA POINTS FOR DEPENDENT VARIABLE.
C
C       NPTS - NUMBER OF DATA POINTS.
C
C       NTERMS - NUMBER OF PARAMETERS (OR INDEP VAR'S) TO BE OPTIMIZED.
C
C       A - ARRAY OF PARAMETERS (OR INDEP VAR'S) TO BE OPTIMIZED.
C
C       DELTAA - ARRAY OF INCREMENTS FOR PARAMETERS A (FOR USE IN
C                DERIV).
C
C       SIGMAA - ARRAY OF STANDARD DEVIATIONS FOR PARAMETERS A.
C
C       FLAMDA - PROPORTION OF GRADIENT SEARCH INCLUDED.
C
C       YFIT - ARRAY OF CALCULATED VALUES OF Y.
C
C       CHISQR - REDUCED CHI SQUARE FOR FIT.
C
C       FNCTN - NAME OF SUBROUTINE USED TO CALCULATE THE FUNCTION
C               VALUES.
C
C       DERIV,Y1,Y2 - WORK ARRAYS.
C
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C        FNCTN (OR EQUIVALENT WITH A DIFFERENT NAME)
C        FDERIV  MATINV  FCHISQ
C
C     COMMENTS
C        DIMENSION STATEMENT VALID FOR NTERMS UP TO 64
C        SET FLAMDA=.001 AT BEGINNING OF SEARCH
C        CONTROL TRANSFERRED BACK TO CALLING PROG AFTER EACH ITERATION
C
C     ******
C
C        UPDATE DIMENSIONS IN FDERIV WHEN CHANGING NDIM OR MAXPTS
C
C     ******
C
      SUBROUTINE CURFIT (X,Y,NPTS,NTERMS,A,DELTAA,SIGMAA,FLAMDA,YFIT,
     *CHISQR,FNCTN,DERIV,Y1,Y2)
      EXTERNAL FNCTN
      DOUBLE PRECISION ARRAY,A,YFIT,ALPHA,BETA,DERIV,B,Y1,Y2,DSQRT
      DIMENSION X(NPTS),SIGMAA(NTERMS),Y(NPTS),A(NTERMS),DELTAA(NTERMS),
     *YFIT(NPTS),Y1(NPTS),Y2(NPTS),DERIV(NPTS,NTERMS)
      DIMENSION ALPHA(20,20),BETA(20),ARRAY(20,20),B(20),IK(20),JK(20)
      DATA NDIM/20/
      IF (NTERMS .LE. NDIM)  GO TO 11
      WRITE (6,10)  NTERMS
```

```
   10 FORMAT (//' NTERMS=',I6,4X,'TOO LARGE IN CURFIT')
      STOP
   11 NFREE = NPTS - NTERMS
      IF (NFREE)  13,13,20
   13 NFREE = 1
C
C        EVALUATE ALPHA AND BETA MATRICES
C
   20 DO 34  J = 1,NTERMS
      BETA(J) = 0.D0
      DO 34  K = 1,J
   34 ALPHA(J,K) = 0.D0
      CALL FNCTN (X,YFIT,NPTS,A,NTERMS)
      CALL FDERIV (X,NPTS,A,DELTAA,NTERMS,DERIV,FNCTN,Y1,Y2)
      DO 50  I = 1,NPTS
      DO 46  J = 1,NTERMS
      BETA(J) = BETA(J) + (Y(I) - YFIT(I)) * DERIV(I,J)
      DO 46  K = 1,J
   46 ALPHA(J,K) = ALPHA(J,K) + DERIV(I,J) * DERIV(I,K)
   50 CONTINUE
   51 DO 53  J = 1,NTERMS
      DO 53  K = 1,J
   53 ALPHA(K,J) = ALPHA(J,K)
C
C        EVALUATE CHI SQUARE AT STARTING POINT
C        INVERT MODIFIED CURVATURE MATRIX TO FIND NEW PARAMETERS
C
      CHISQ1 = FCHISQ(Y,YFIT,NPTS,NFREE)
   71 DO 74  J = 1,NTERMS
      DO 73  K = 1,NTERMS
   73 ARRAY(J,K) = ALPHA(J,K) / DSQRT(ALPHA(J,J) * ALPHA(K,K))
   74 ARRAY(J,J) = 1.D0 + FLAMDA
   80 CALL MATINV (ARRAY,NTERMS,DET,IK,JK,NDIM)
   81 DO 84  J = 1,NTERMS
      B(J) = A(J)
      DO 84  K = 1,NTERMS
   84 B(J) = B(J) + BETA(K) * ARRAY(J,K)/DSQRT(ALPHA(J,J) * ALPHA(K,K))
C
C        IF CHI SQUARE INCREASED, INCREASE FLAMDA AND TRY AGAIN
C
   91 CALL FNCTN (X,YFIT,NPTS,B,NTERMS)
   93 CHISQR = FCHISQ(Y,YFIT,NPTS,NFREE)
      IF (CHISQ1 - CHISQR)  95,101,101
   95 FLAMDA = 10. * FLAMDA
      GO TO 71
C
C        EVALUATE PARAMETERS AND UNCERTAINTIES
C
  101 DO 103  J = 1,NTERMS
      A(J) = B(J)
  103 SIGMAA(J) = SQRT(SNGL(ARRAY(J,J) * CHISQR/ALPHA(J,J)))
      FLAMDA = FLAMDA / 10.
  110 RETURN
      END
```

```
C
C     SUBROUTINE MATINV
C
C     PURPOSE
C        INVERT A SYMMETRIC MATRIX AND CALCULATE ITS DETERMINANT.
C
C     DESCRIPTION OF PARAMETERS:
C     ARRAY  - REAL*8 INPUT MATRIX WHICH IS REPLACED BY ITS INVERSE.
C     NORDER - DEGREE OF MATRIX (ORDER OF DETERMINANT).
C     DET    - REAL*4 DETERMINANT OF INPUT MATRIX (RETURNED BY IRRFIT).
C     IK,JK  - INTEGER*4 WORK ARRAYS (LENGTH NORDER).
C     NDIM   - DIMENSION OF ARRAY IN CALLING PROGRAM. ONLY THE PORTION
C              NORDER BY NORDER WILL BE INVERTED.
C
      SUBROUTINE MATINV (ARRAY, NORDER, DET, IK, JK, NDIM)
C
      INTEGER*4 IK(NORDER), JK(NORDER)
C
      REAL*8 AMAX, SAVE, ARRAY(NDIM,NDIM), DABS
C
C
   10 DET = 1.0
   11 DO 100 K=1,NORDER
C
C     FIND LARGEST ELEMENT ARRAY(I,J) IN REST OF MATRIX.
C
      AMAX = 0.0D0
   21 DO 30 I=K,NORDER
      DO 30 J=K,NORDER
   23 IF (DABS(AMAX)-DABS(ARRAY(I,J))) 24,24,30
   24 AMAX = ARRAY(I,J)
      IK(K) = I
      JK(K) = J
   30 CONTINUE
C
C     INTERCHANGE ROWS AND COLUMNS TO PUT AMAX IN ARRAY(K,K).
C
   31 IF (AMAX) 41,32,41
   32 DET = 0.0
      GO TO 140
   41 I = IK(K)
      IF (I-K) 21,51,43
   43 DO 50 J=1,NORDER
      SAVE = ARRAY(K,J)
      ARRAY(K,J) = ARRAY(I,J)
   50 ARRAY(I,J) = -SAVE
   51 J = JK(K)
      IF (J-K) 21,61,53
   53 DO 60 I=1,NORDER
      SAVE = ARRAY(I,K)
      ARRAY(I,K) = ARRAY(I,J)
   60 ARRAY(I,J) = -SAVE
C
C     ACCUMULATE ELEMENTS OF INVERSE MATRIX.
C
   61 DO 70 I=1,NORDER
      IF (I-K) 63,70,63
   63 ARRAY(I,K) = -ARRAY(I,K) / AMAX
   70 CONTINUE
   71 DO 80 I=1,NORDER
```

```
      DO 80 J=1,NORDER
      IF (I-K) 74,80,74
   74 IF (J-K) 75,80,75
   75 ARRAY(I,J) = ARRAY(I,J) + ARRAY(I,K)*ARRAY(K,J)
   80 CONTINUE
   81 DO 90 J=1,NORDER
      IF (J-K) 83,90,83
   83 ARRAY(K,J) = ARRAY(K,J) / AMAX
   90 CONTINUE
      ARRAY(K,K) = 1.0D0 / AMAX
  100 DET = DET * AMAX
C
C     RESTORE ORDERING OF MATRIX.
C
  101 DO 130 L=1,NORDER
      K = NORDER - L + 1
      J = IK(K)
      IF (J-K) 111,111,105
  105 DO 110 I=1,NORDER
      SAVE = ARRAY(I,K)
      ARRAY(I,K) = -ARRAY(I,J)
  110 ARRAY(I,J) = SAVE
  111 I = JK(K)
      IF (I-K) 130,130,113
  113 DO 120 J=1,NORDER
      SAVE = ARRAY(K,J)
      ARRAY(K,J) = -ARRAY(I,J)
  120 ARRAY(I,J) = SAVE
  130 CONTINUE
  140 RETURN
      END
```

```
C
C     SUBROUTINE FDERIV
C
C     PURPOSE
C        EVALUATE PARTIAL DERIVATIVES OF AN ARBITRARY FUNCTION AT
C        A SET OF POINTS.
C
C     DESCRIPTION OF PARAMETERS:
C
C       X - ARRAY OF POINTS AT WHICH ALL PARTIAL DERIVATIVES ARE
C           REQUIRED.
C
C       NPTS - NUMBER OF POINTS AT WHICH ALL PARTIAL DERIVATIVES ARE
C              REQUIRED.
C
C       V - ARRAY OF INDEPENDENT VARIABLES (OR PARAMETERS) WITH RESPECT
C           TO WHICH PARTIAL DERIVATIVES ARE TO BE EVALUATED.
C
C       DELTAV - ARRAY OF INCREMENTS OF THE INDEPENDENT VARIABLES TO BE
C                USED IN DIFFERENCE EQUATION APPROX TO DERIVATIVES.
C
C       NVAR - NUMBER OF INDEPENDENT VARIABLES.
C
C       DERIV - ARRAY OF CALCULATED PARTIAL DERIVATIVES.
C
C       FNCTN - NAME OF SUBROUTINE USED TO CALCULATE THE FUNCTION.
C
C     REQUIRED SUBROUTINE
C        FNCTN (OR EQUIVALENT WITH A DIFFERENT NAME)
C
      SUBROUTINE FDERIV (X,NPTS,V,DELTAV,NVAR,DERIV,FNCTN,Y1,Y2)
      DOUBLE PRECISION V,DERIV,Y1,Y2,VJ,DELTA,DABS,D
      DIMENSION X(NPTS),V(NVAR),DELTAV(NVAR),DERIV(NPTS,NVAR),Y1(NPTS),
     *Y2(NPTS)
      DO 100  J = 1,NVAR
      VJ = V(J)
      DELTA = DELTAV(J)
      V(J) = VJ + DELTA
      CALL FNCTN (X,Y1,NPTS,V,NVAR)
      V(J) = VJ - DELTA
      CALL FNCTN (X,Y2,NPTS,V,NVAR)
      DO 50  I = 1,NPTS
      DERIV(I,J) = (Y1(I) - Y2(I)) / (2.D0*DELTA)
      D = DERIV(I,J)
      IF(DABS(D).GT.1.D10) WRITE(6,40) I,J,Y1(I),
     1 Y2(I),V(J),DELTA,DERIV(I,J)
   40 FORMAT(' OVER-FLOW IN FDERIV: I,J,Y1(I),Y2(I),V(J),DELTA,DER
     1IV(I,J)='/2I5,5D15.7)
      IF(DABS(D) .LT. 1.D-10)  DERIV(I,J) = 0.D0
      IF(DABS(D) .GT. 1.D10)  DERIV(I,J) =(D/DABS(D)) * 1.D10
   50 CONTINUE
      IF (DERIV(J,J) .EQ. 0.D0)DERIV(J,J) = 1.D-10
  100 V(J) = VJ
      RETURN
      END
```

```
C
C     FUNCTION FCHISQ
C
C     PURPOSE
C        EVALUATE REDUCED CHISQUARE FOR FIT TO DATA.
C
C     DESCRIPTION OF PARAMETERS:
C
C       Y - ARRAY OF DATA POINTS.
C
C       YFIT - ARRAY OF CALCULATED VALUES OF Y.
C
C       NPOINTS - NUMBER OF DATA POINTS.
C
C       NFREE - NUMBER OF DEGREES OF FREEDOM.
C
      FUNCTION FCHISQ (Y,YFIT,NPTS,NFREE)
      DOUBLE PRECISION CHISQ,YFIT
      DIMENSION Y(NPTS),YFIT(NPTS)
      CHISQ = 0.
      IF (NFREE)  10,10,20
   10 FCHISQ = 0.
      RETURN
C
C       ACCUMULATE CHI SQUARE AND DIVIDE BY NO OF DEG'S OF FREEDOM
C
   20 DO 30  I = 1,NPTS
   30 CHISQ = CHISQ + (Y(I) - YFIT(I)) ** 2
C     WRITE(6,40) (I,Y(I),YFIT(I),I=1,NPTS)
C  40 FORMAT(2(' I=',I5,' Y,YFIT=',2E15.5,5X))
      FREE = NFREE
      FCHISQ = CHISQ / FREE
      RETURN
      END
```

```
Listing of FNCTN.FOR,  7-Jan-1988  11:14:38                                Page 1

C
C     SUBROUTINE FNCTN
C
C     TO CALCULATE THE FITTED CURVE
C
      SUBROUTINE FNCTN(X,YFIT,NPTS,V       ,NTERMS)
C
      REAL*4    X(NPTS)
      REAL*8    YFIT(NPTS),V(NTERMS),WIDTH,AMPL,MEAN1
      DO 100 I=1,NPTS
C100  YFIT(I) =   V(2)*DEXP(-.5D0*((X(I)-  V(1))/  V(3))
C    +**2)/V(3)+V(5)*DEXP(-.5D0*((X(I)-V(4))/V(6))**2)/V(6)+V(7)
 100  YFIT(I) = 0.159154943 * V(2) * V(3)/((X(I)-V(1) )**2.+(V(3)/2.)
     +**2.)+ 0.159154943 * V(5) * V(6)/((X(I)-V(4))**2.+ (V(6) /2.)**2.)
     ++ V(7) + V(8) * X(I)
      RETURN
      END
```

```
C
C      DDATA.FOR
C
C      This program dumps a data file which was written by COAL.C or
C      FAKE.C. It is printed out in 6 pairs of columns. Each pair
C      consists of the point number followed by the actual datum.
C
       REAL * 4  Y(512), samwt
       CHARACTER * 18  filename
C
C      Read in the data at this point.
C
       call rdfile(NPTS,Y,filename,samwt)
c      do 80 K = 1,NPTS
c  80  write (0,100) K, Y(K)
c 100  format (I7,F7.0)
       nrows = aint ( npts/6.0 + 0.9 )
       n = nrows
       do 150 I = 1,n
  150  write (*,200)I,Y(I),I+N,Y(I+N),I+2*N,Y(I+2*N),I+3*N,Y(I+3*N),
      +I+4*N,Y(I+4*N),I+5*N,Y(I+5*N)
  200  format (I3, F6.0, 5(I8,F6.0))
       end
```

```
Listing of AREA.FOR,  19-Aug-1988  13:41:18                        Page 1

C
C     AREA
C
      real * 4    area, baseline, total, Y(512),samwt,pkht
      character * 18  filename

C
C     Read in the desired file
C
      call rdfile (NPTS, Y, filename, samwt)
C
C     Calculate the baseline as the average over the first 40 points
C
      total = 0.0
      do 100 K = 1,40
 100  total = total + Y(K)
      K = K - 1
      baseline = total/K
      write (*,200) K, baseline
 200  format (´ The average baseline over the first ´,I3,
     +´ points is ´,F8.3)
C
C     Print the accumulated area at every point and find the
C     maximum peak height wrt baseline.
C
      area = 0.0
      pkht = 0.0
      do 400 I = 41,NPTS
      if (Y(I) - baseline .gt. pkht) pkht = Y(I) - baseline
      area = area + Y(I) - baseline
      write (*,300) I, Y(I), area
 300  format (2X, I4, F12.1, F15.3)
 400  continue
      write (*,500) pkht
 500  format (2X,´The peak height is ´, F8.1)
      end
```

```
C
C     SUBROUTINE WRCAL
C
C     This subroutine writes the multiplication factor to the unique
C     file \data\mult.fac


      SUBROUTINE WRCAL (multfac)

      REAL * 4    multfac
      REAL * 8    calfac

      open (2, file = '\data\mult.fac', recl = 4,
     +form = 'UNFORMATTED', access = 'DIRECT', mode = 'WRITE',
     +err = 9000)
  200 write (2, rec = 1) multfac
      close (2)
      go to 5000
 9000 write (*,9010)
 9010 format (' WRFILE: Can''t open that file.')
 5000 return
      end
```

```
C
C     CAL PROGRAM
C
      real * 4 apergram, total, multfac
      real * 8  calfac
      integer * 2  nfiles
      character * 1  yesno

      call rdcal(multfac)
      write (*,30) multfac
   30 format (´ The current multiplier is ´,f15.6)
      write (*,100)
  100 format (´ How many files will be used for calibration?´)
      read (*,200) nfiles
      if (nfiles .eq. 0) goto 900
  200 format (I4)
      total = 0.0
      do 300 I = 1, nfiles
      call areasub (apergram)
      write (*,250) apergram
  250 format (´ This file has ´,e12.6,´ area units per gram.´)
  300 total = total + apergram
      calfac = total/nfiles
      multfac = 40000000/calfac
      write (*,400) calfac
  400 format (´ The calibration factor is ´,e12.6,´ area units´,
     +´ per gram.´)
      write (*,450) multfac
  450 format (´ The multiplier is ´,f11.7)
      write (*,500)
  500 format (´ Do you want to put this calibration into effect? (y/n)´)
      read (*,600) yesno
  600 format (A1)
      if (yesno .eq. ´Y´ .or. yesno .eq. ´y´) then
            goto 800
      else
            goto 900
      endif
  800 call wrcal(multfac)
  900 end
```

```
C
C     SUBROUTINE RDCAL
C
C     This subroutine reads the multiplication factor from the unique
C     file \data\mult.fac. It is used to multiply the amplitude of
C     each point above the baseline.

      SUBROUTINE RDCAL (multfac)

      REAL * 4    multfac
      REAL * 8    calfac

      open (2, file = '\data\mult.fac', recl = 4,
     +form = 'UNFORMATTED', access = 'DIRECT', mode = 'READ',
     +err = 9000)
  200 read (2, rec = 1) multfac
      close (2)
      go to 5000
 9000 write (*,9010)
 9010 format (' RDCAL: Can''t open that file.')
 5000 return
      end
      SUBROUTINE WRADJ (N,YADJ,filename,samwt)
c
c     This subroutine writes a file that has N points of adjusted
c     data, YADJ(I). Each real point is converted to INT*2 before
c     being written to the file. The written file has the same format
c     as the files stored by the data acquisition software. The name
c     of the written file is the filename passed to this subroutine
c     with an 'a' appended to its beginning.
c     The file is written in directory \data.
c
      INTEGER * 2  X(512)
      REAL * 4     YADJ(512),samwt
      CHARACTER * 18 filename

c
c     ************ write the sample weight ***************
c
      open (1,file = '\data\a'// filename, recl=6,
     +form='UNFORMATTED', access='DIRECT',mode = 'WRITE',
     +err = 9000)
      write (1,rec=2) samwt
 1000 close(1)
c
      do 100 I = 1,N
  100 X(I) = INT2 (YADJ(I))
      open (2, file = '\data\a'// filename, recl = 2,
     +form = 'UNFORMATTED', access = 'DIRECT', mode = 'WRITE',
     +err = 9000)
      do 200 I = 1,N
  200 write (2, rec = 3 * I + 7) X(I)
      close (2)
      go to 5000
 9000 write (*,9010)
 9010 format (' WRADJ: Can''t open that file.')
 5000 return
      end
```

```
C
C     LINEAR
C
C     This program takes every point in a spectrochronogram and adjusts
C     it. This adjustment is a linearization to compensate for the
C     non-linearity of the detector. A calibration exists which is a
C     plot of peak height (voltage) versus weight of sulfur (in
C     benzenesulfonic acid). This calibration is made up of 15 straight
C     line segments, each with a known slope and intercept. There is a
C     single straight line with slope M1 and intercept B1 which covers
C     the same range of sulfur weights as the 15 segments. Every point
C     y2 on one of the 15 segments (with slope M2 and intercept B2) is
C     converted to the corresponding point y1 (at the same abscissa) on
C     the single straight line with the formula:
C
C                 y1 = ( M1 / M2 ) * ( y2 - B2 ) + B1
C
C     Observed voltages are measured with respect to the baseline,
C     which is taken as the average of the first 40 points in the
C     spectrochronogram. Adjusted voltages are added onto the baseline.
C     The adjusted spectrochronogram is stored in a new file whose name
C     is the same as the input file with an 'a' appended to the
C     beginning.

      real * 4    area, baseline, total, Y(512), YADJ(512), samwt,pkht
      real * 4    M1, M2, B1, B2, WY
      character * 18  filename

      M1 = 611.6716
      B1 = 32.4846

C
C     Read in the desired file
C
      call rdfile (NPTS, Y, filename, samwt)

C
C     Calculate the baseline as the average over the first 40 points
C
      total = 0.0
      do 100 K = 1,40
 100  total = total + Y(K)
      K = K - 1
      baseline = total/K
      write (*,200) K, baseline
 200  format (' The average baseline over the first ',I3,
     +' points is ',F8.3)
C
C     Adjust each point
C
      do 300 I = 41, NPTS
      WY = Y(I) - baseline
      if ( WY .lt. 30.80 ) then
            B2 = -0.0607
            M2 = 308.638
      elseif ( WY .lt. 78.79 ) then
            B2 = -19.817
            M2 = 493.046
      elseif ( WY .lt. 139.30 ) then
            B2 = -42.30
```

```
                M2 = 605.333
        elseif ( WY .lt. 271.33 ) then
                B2 = -58.75
                M2 = 660.167
        elseif ( WY .lt. 427.639 ) then
                B2 = -119.005
                M2 = 780.920
        elseif ( WY .lt. 623.99 ) then
                B2 = -25.2128
                M2 = 649.206
        elseif ( WY .lt. 978.12 ) then
                B2 = -84.5403
                M2 = 708.442
        elseif ( WY .lt. 1360.14 ) then
                B2 = -176.234
                M2 = 768.185
        elseif ( WY .lt. 1703.22 ) then
                B2 = 12.1645
                M2 = 676.421
        elseif ( WY .lt. 2017.67 ) then
                B2 = 161.603
                M2 = 618.689
        elseif ( WY .lt. 2323.31 ) then
                B2 = 217.753
                M2 = 601.587
        elseif ( WY .lt. 2557.77 ) then
                B2 = 853.834
                M2 = 425.985
        elseif ( WY .lt. 2797.55 ) then
                B2 = 657.481
                M2 = 475.572
        elseif ( WY .lt. 3100.95 ) then
                B2 = 12.129
                M2 = 617.765
        else
                B2 = 1102.873
                M2 = 404.667
        endif
        YADJ(I) =  ( M1/M2 ) * ( WY - B2 ) + B1  + baseline
C       write ( *,250 ) I, WY, YADJ(I)
 250    format (2x,I3,'    old = ',f10.3,'       new = ',f10.3)
 300    continue
        do 400 I = 1,40
 400    YADJ(I) = Y(I)
        call wradj (NPTS,YADJ,filename,samwt)
        end
```

```
C
C     ADJUST
C
C     This program takes every point in a spectrochronogram and adjusts
C     it. This adjustment is a linearization to compensate for the
C     non-linearity of the detector. A calibration exists which is a
C     plot of peak height (voltage) versus weight of sulfur (in
C     benzenesulfonic acid). This calibration is made up of 15 straight
C     line segments, each with a known slope and intercept. There is a
C     single straight line with slope M1 and intercept B1 which covers
C     the same range of sulfur weights as the 15 segments. Every point
C     y2 on one of the 15 segments (with slope M2 and intercept B2) is
C     converted to the corresponding point y1 (at the same abscissa) on
C     the single straight line with the formula:
C
C                 y1 = ( M1 / M2 ) * ( y2 - B2 ) + B1
C
C     Observed voltages are measured with respect to the baseline,
C     which is taken as the average of the first 40 points in the
C     spectrochronogram. Once a voltage has been linearized, it is
C     multiplied by the multiplication factor, multfac. Adjusted voltages
C     are added onto the baseline. The adjusted spectrochronogram is stored
C     in a new file whose name is the same as the input file with an 'a'
C     appended to the beginning.

      real * 4    area, baseline, total, Y(512), YADJ(512), samwt,pkht
      real * 4    M1, M2, B1, B2, WY, multfac
      character * 18  filename

      M1 = 611.6716
      B1 = 32.4846

C
C     Read in the desired file
C
      call rdfile (NPTS, Y, filename, samwt)

C
C     Get the multiplication factor
C
      call rdcal ( multfac )

C
C     Calculate the baseline as the average over the first 40 points
C
      total = 0.0
      do 100 K = 1,40
 100  total = total + Y(K)
      K = K - 1
      baseline = total/K
      write (*,200) K, baseline
 200  format (' The average baseline over the first ',I3,
     +' points is ',F8.3)
C
C     Adjust each point
C
      do 300 I = 41, NPTS
      WY = Y(I) - baseline
      if ( WY .lt. 30.80 ) then
           B2 = -0.0607
```

```
                  M2 = 308.638
            elseif ( WY .lt. 78.79 ) then
                  B2 = -19.817
                  M2 = 493.046
            elseif ( WY .lt. 139.30 ) then
                  B2 = -42.30
                  M2 = 605.333
            elseif ( WY .lt. 271.33 ) then
                  B2 = -58.75
                  M2 = 660.167
            elseif ( WY .lt. 427.639 ) then
                  B2 = -119.005
                  M2 = 780.920
            elseif ( WY .lt. 623.99 ) then
                  B2 = -25.2128
                  M2 = 649.206
            elseif ( WY .lt. 978.12 ) then
                  B2 = -84.5403
                  M2 = 708.442
            elseif ( WY .lt. 1360.14 ) then
                  B2 = -176.234
                  M2 = 768.185
            elseif ( WY .lt. 1703.22 ) then
                  B2 = 12.1645
                  M2 = 676.421
            elseif ( WY .lt. 2017.67 ) then
                  B2 = 161.603
                  M2 = 618.689
            elseif ( WY .lt. 2323.31 ) then
                  B2 = 217.753
                  M2 = 601.587
            elseif ( WY .lt. 2557.77 ) then
                  B2 = 853.834
                  M2 = 425.985
            elseif ( WY .lt. 2797.55 ) then
                  B2 = 657.481
                  M2 = 475.572
            elseif ( WY .lt. 3100.95 ) then
                  B2 = 12.129
                  M2 = 617.765
            else
                  B2 = 1102.873
                  M2 = 404.667
            endif
            YADJ(I) =  multfac * ( ( M1/M2 ) * ( WY - B2 ) + B1 ) + baseline
C           write ( *,250 ) I, WY, YADJ(I)
 250        format (2x,I3,'    old = ',f10.3,'      new = ',f10.3)
 300        continue
            do 400 I = 1,40
 400        YADJ(I) = Y(I)
            call wradj (NPTS,YADJ,filename,samwt)
            end
```

```
/*      file:   acqfree.c

        LPSCHART -- STRIP CHART RECORDING UTILITY PROGRAM

    ==============================================================================
    |                                                                            |
    |  Copyright (C) 1987.    Data Translation, Inc.   100   Locke    Drive,     |
    |  Marlboro Massachusetts 01752-1192.                                        |
    |                                                                            |
    |  The information in this document is subject to change  without  notice   |
    |  and  should  not  be  construed  as a  commitment by Data Translation,    |
    |  Inc.  Data  Translation, Inc.  does  not  assume   any  responsibility    |
    |  for any errors that may appear in this document.                          |
    |                                                                            |
    |  This document is provided solely as an example of how to utilize  Data    |
    |  Translation   software   libraries  in  a  sample  application.   Data    |
    |  Translation, Inc.  does  not  assume  any   responsibility   for   the    |
    |  use   or  reliability  of any portion of this document or the described   |
    |  software.                                                                 |
    |                                                                            |
    |  General permission to copy or modify, but not for  profit,  is  hereby    |
    |  granted,  provided  that  the  above  copyright notice is included and    |
    |  reference made to the fact that reproduction privileges  were  granted    |
    |  by Data Translation, Inc.                                                 |
    |                                                                            |
    ==============================================================================

    The strip chart recording program provides a means for users to acquire
    analog-to-digital (A/D) data and display that data in the form of a strip
    chart.

    The strip chart recording program is divided into two modules: this module
    which contains the routines which acquire the A/D data and display that data in
    the form of a strip chart; and the screenio module which contains the routines
    for creating the physical appearance of data on the display monitor such as
    highlighted text, blinking text, and the drawing of lines and dots.

    This program was designed to be used with LPCLAB and a DT2811 or DT2814 data
    acquisition board configured for differential A/D input.  The DT2811 or DT2814
    (whichever is being used) must be configured for an input voltage range of +5
    to -5 volts.

    This program can be compiled and linked using MICROSOFT C V4.0 with the
    following commands:

    COMPILATION COMMAND LINE --
            MSC /GO /Od /AS LPSCHART.C;
            MSC /GO /AS SCREENIO.C;

    LINK COMMAND LINE --
            LINK LPSCHART+SCREENIO,,,LPCLCS.LIB;


    Note: The functions in the module SCREENIO use some of the non-standard
          C run-time library functions as well as several standard C run-time
          library functions.  The non-standard run-time library functions may not
          be supported in future releases of MicroSoft's C compiler.
```

```
/* Enable argument type checking on library functions */

#define         LINT_ARGS       1

/* Include necessary C run-time library header files */

#include        <string.h>
#include        <time.h>
#include        <stdio.h>

/* Include header files created for this utility program */

#include        "utility.h"
#include        "screenio.h"

/* Define character size in pixels for a 40 column display */

#define         CHAR_HEIGHT     8
#define         CHAR_WIDTH      8

int     col_end = 0;    /* # of columns on screen minus 1 (zero relative) */

/* Data storage for header information of log files being read */
struct  log_file_header         in_hdr = { 0 };

/* Data translation header information of log files being written*/
struct  log_file_header         out_hdr =
{
        'D','T','I',':','L','P','C','L','A','B',
        ' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',
        ' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',
        ' ',' ',
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0
};

/* Data file header for Leco sampling*/

struct  sample_file_header      samp_hdr={0,0,0,0,0};
char            line[ LINE_LEN + 1];    /* Store characters in line buffer.  */
```

```
main()
{
/*
 This is the strip chart main function which dispatches to various other
 functions to perform the actual work.  Refer to the functions called
 by this routine for more details.
*/
        /*
         Check to see if the device driver ANSI.SYS is loaded and
         if so warn the user about its affect on the appearance of
         the screen.
        */

        if( report_installed_video_driver() == ANSI_DEVICE_MODE )
        {
                print_ansi_device_warning();
        }

        /*
         Initialize the video state and screen characteristics:
           white foreground, blue background, low resolution color
           graphics mode with 40 column text.
        */

        if( initialize_video( WHITE, BLUE, CO_LOW_RES) == BW_CARD )
        {  /*  If no CGA, then print message and return to DOS */
                print_cga_missing_warning();
                return;
        }

        /*
         Check to see if we are using an 40 column display and if
         not warn the user that this program was written to be used
         on an 40 column display.
        */

        if( (col_end = (report_display_column_size() - 1)) != (40 - 1) )
        {
                print_column_size_warning();
        }

        /*
         Display the main menu and allow the user to select from the options
         provided.  Then dispatch to the appropriate item selection function
         handler.
        */

        main_menu();

        /*
         Return the video characteristics as much as possible to
         the state that it was in before this program started.
        */

        restore_video_state();

        /* The end! */

        return;
} /* main */
```

```
main_menu()
{
 /*
 This routine displays the menu of options and allows the user to select one
 of the options.  Once a valid option has been selected, the appropriate
 action function which handles that option is called.

 The only ways to exit to DOS is by selecting the exit item from the menu or
 by pressing the CNTRL-C key combination.  Selecting the exit item is the
 desired method because this will allow for proper termination such as
 closing opened files.  The CNTRL-C method simply aborts to DOS.  Note that
 pressing the <ESC> key causes the same affect as selecting the exit item
 from the menu.
*/

        int                     i;                      /* Loop control variable.          *
        int                     ch;                     /* Input character from keyboard.  *
        int                     select;                 /* Menu item selection.            *
        int                     got_file;               /* Indicates valid file name found. *
        int                     item_count;             /* # of items to write into the file *
        int             items_write;            /* # of items actually written      */
        int                     items_read;
        FILE            *stream;                /* Open file structure pointer.     */

        /*
         Continue allowing the user to select items from the main menu and go
         off and perform the selected action item.  Once the action is
         complete, repeat the main menu selection process over and over again
         until the user selects to exit. Upon selection of the exit item (or
         escape key), control immediately returns to the calling program and
         we exit from this loop.
        */

        while( TRUE )
        {

        /* Clear the entire screen */

                clear_text_window( 0, 0, 24, col_end, NORMAL_VIDEO);

        /* Display the title */

                clear_text_window( 0, col_end, 3, col_end, BOLD_VIDEO);
                position_cursor( 0, 12);
                printf("DATA TRANSLATION");
                position_cursor( 1, 19);
                printf("***");
                position_cursor( 3, 6);
                printf("SULPHUR WAVEFORM RECORDING");

        /* Display the menu */

                position_cursor( 6, 0);
                printf(" The following options are available:\n");
                printf("\n");
                printf("       1 -- Chart and store newly");
                printf("                     acquired data,\n");
                printf("       2 -- Chart data from a previously");
                printf("              logged data file,\n");
        printf("       3 -- Exit.\n");
        printf("\n");
        printf(" Select an option (1, 2, or 3): ");

/* Select a menu item */

        ch = getch();
        while( (ch < '1') || (ch > '3') )
        { /* Screen out invalid selections.  Also, check for ESC key */
                if( ch == ESC )
                {
                        return;
                }
                BEEP;
                ch = getch();
        }

/* Indicate which item was selected */

        select = ch - '0';
        if( select > 2 )
        {
                clear_text_window( (8 + select), 0, (8 + select),
                                5, REVERSE_AND_BLINK_VIDEO);
                position_cursor( (8 + select), 0);
        }
        else
        {
                clear_text_window( (7 + select), 0, (7 + select),
                                5, REVERSE_AND_BLINK_VIDEO);
                position_cursor( (7 + select), 0);
        }
        printf("=====>");


/* Check for selection of exit item and exit if found */

        if( select == 3 )
        {  /* Exit item found.  Print exit message for a time period */
                long    timer;

                position_cursor( 16, 0);
                printf("EXITING!");
                for( timer = TIME_DELAY; timer > 0; timer--);
                return;
        }

        if( select == 1 )
        {
                clear_text_window(0,0,16,79,NORMAL_VIDEO);
                position_cursor(0,0);
                printf(">Sample number(1 TO 65535):");
                scanf("%d",&samp_hdr.sample_number);
                printf("\n>Run number:");
                scanf("%d",&samp_hdr.run_number);
                printf("\n>Sample weight:");
                scanf("%f",&samp_hdr.sample_weight);
                printf("\n>Total Sulphur Content:");
                scanf("%f",&samp_hdr.sulphur_content);
                got_file = FALSE;
        }
                if(select ==1 || select ==2)
                {
                got_file = FALSE;
/* Keep asking the user for a file name until it's valid */

                while( !got_file )
                {

        /* Prompt the user to enter a file name */

                        position_cursor( 16, 0);
                        printf(" Enter name of disk log file\n");
                        printf("  (e.g. C:\\DATA\\VOLTS.LOG):\n");
                        clear_text_window( 18, 3, 18,
                                        col_end, NORMAL_VIDEO);
                        clear_text_window( 18, 0, 18, 2, BLINK_VIDEO);
                        position_cursor( 18, 0);
                        printf(" > ");
                        flushall();     /* Clear keyboard input */

        /* Input a line (the file name) from keyboard */

                        i = 0;
                        ch = getche();
                        while( (ch != EOF) && (ch != LF)
                                && (ch != CR) && (i < LINE_LEN + 1) )
                        {
                                if( ch == ESC )
                                {
                                         return;         /* Exit on ESC key */
                                }
                                else if( ch == BS )
                                { /* Back space on the back space key */
                                        if( i > 0 )
                                        {
                                                i--;
                                                printf(" \010");
                                        }
                                        else
                                        {
                                                printf(" ");
                                        }
                                }
                                else
                                { /* Save the character in the line buffer */
                                        line[ i++] = ( char )ch;
                                }
                                ch = getche();
                        }
                        line[ i] = NULL;

        /* We have a file name; turn off the blinking prompt */

                        clear_text_window( 18, 0, 18, 2, NORMAL_VIDEO);
                        position_cursor( 18, 0);
                        printf(" > ");

                        clear_text_window( 19, 0, 23, col_end, NORMAL_VIDEO);
        /*
         If the named file doesn't exist, then we must ask for another
         file name -- this operation requires an existing file.
        */

                        if(select == 1)
                        {
                                stream = fopen( &line[ 0], "w+b");
                        }
                        else
                        {
                                stream = fopen( &line[0], "rb");
                        }
                        got_file = TRUE;
                        if( stream == NULL )
                        {
                                position_cursor( 20, 1);
                                printf("The file you specified,\n");
                                printf("  '%s'", &line[0]);
                                position_cursor( 22, 1);
                                printf("can't be found. Try again!\n");
                                printf("  (press the ESC key to quit)");
                                got_file = FALSE;
                        }
                        else
                        {
         item_count = SAMPLE_FILE_HEADER_SIZE / sizeof( int);
         if(select == 1)
         {
         items_write = fwrite((char *)&samp_hdr,
                         sizeof( int), item_count, stream);
         if( items_write != item_count )
         {/* If entire header couldn't be read, print message */
                 position_cursor( 19, 0);
                 printf("item_count=%d",item_count);
                 printf("\nitems_write = %d",items_write);
                 printf("Can't write the header");
                 while( (ch = getch()) != ESC );
                 return;
         }
         }
         else
         {

                 items_read = fread((char*)&samp_hdr,sizeof(int),
                                 item_count,stream);
                 if( items_read != item_count )
                 {
                 position_cursor( 19, 0);
                 printf("item_count=%d",item_count);
                 printf("\nitems_read = %d",items_read);
                 printf("Can't read the header");
                 while( (ch = getch()) != ESC );
                 return;
                 }

     clear_text_window(0,0,16,79,NORMAL_VIDEO);
         position_cursor(0,0);
         PRINTF(">Sample number:%d",samp_hdr.sample_number);
         PRINTF("\n>Run number:%d",samp_hdr.run_number);
```

```
                PRINTF("\n>Sample weight:%f",samp_hdr.sample_weight);
                PRINTF("\n>Total Sulphur Content:%f",samp_hdr.sulphur_content);
                printf("\n**press ESC to continue.**");
                while((ch = getch()) != ESC);
                }
                }
                }
                        }


        /* Dispatch to the strip chart recording function */
                strip_chart( stream,select);

                if( stream != NULL )
                {  /* Close the file if it was open */
                        fclose( stream);
                }
        }
        return;

} /* main_menu */

strip_chart( stream,select)
int      select;
FILE    *stream;
{
/*
 This function continuously gets A/D data and displays it in strip chart
 fashion as a tracking bar scans left to right across the screen over and
 over again.  Up to a total of 3 channels can be plotted each in its own
 color.  If stream equals NULL, then the information is from newly acquired
 data. Else, the information comes from a previously logged data file with
 file pointer stream.
*/
        int     i, j;                   /* Loop control variables.           */
        int     ch;                     /* Input character from keyboard.    */
        int     x1, y1;                 /* Initial point on line drawn.      */
        int     x2, y2;                 /* Terminating point on line drawn.  */
        int     color_id;               /* Pixel value to draw with.         */
        int     plot_count;             /* # of channels to plot.            */
        int     plot_chan[ 3];          /* Holds channel #s being plotted.   */
        int     start_chan;             /* Starting channel in scan.         */
        int     end_chan;               /* Ending channel in scan.           */
        int     chan_count;             /* Total # of channels to scan.      */
        int     xs, ys;                 /* Upper left corner of actual plot. */
        int     xe, ye;                 /* Lower right corner of actual plot.*/
        int     done;                   /* Tells us when plotting is done.   */
        int     position;               /* Tracks current plotting position. */
        int     old_position;           /* Last plotting position.           */
        int     status;                 /* LPCLAB function return status.    */
        int     timing;                 /* A/D acquisition timing source.    */
        int     unit;                   /* LPCLAB unit specifier.            */
        int     gain;                   /* A/D gain.                         */
        int     item_count;             /* # of items to read in from file.  */
        int     items_read;             /* # of items actually read.         */
        int     eof_found;              /* Tells when end-of-file occurs.    */
        int     input_range;            /* The voltage input range of A/D.   */
        int     adc_data[ 8];           /* A/D acquired data storage.        */
        int     xadc_data[8];           /* Temporary adc_data value          */
        int     tic_counter;            /* Counts interval between tic marks.*/
int     tic_location[ 11];      /* y locations of tic marks.          */
int     tic_cnt;                /* The # of tic marks to make.        */
int     old_value[ 3];          /* Old values to plot.                */
int     new_value[ 3];          /* New values to plot.                */
int     port_data[ 8];          /* Storage for digital data read.     */
char    *prompt[ 3];            /* Character strings for prompting.   */
long    time_val;               /* Date and time binary value.        */
float   freq;                   /* The frequency of A/D operations.   */
int     port_mask;
long value;
int     rec_count=0;
prompt[ 0] = "1st":             /* Initialize the prompt messages.    */
prompt[ 1] = "2nd";
prompt[ 2] = "3rd";




port_mask = 0xFF;
/* Establish the screen corners of the actual plot */

xs = (CHAR_WIDTH * 2) + 2;
ys = CHAR_WIDTH + 1;
xe = 319 - 1;
ye = 199 - (CHAR_WIDTH * 3) - 2;

/* Clear the entire screen */

clear_text_window( 0, 0, 24, col_end, NORMAL_VIDEO);
position_cursor( 0, 0);

/* Initialize the data acquisition parameters */

timing = 0;
unit = 1;
gain = 1;
freq = .5;

/* Display the title */

clear_text_window( 0, col_end, 3, col_end, BOLD_VIDEO);
position_cursor( 0, 12);
printf("DATA TRANSLATION");
position_cursor( 1, 19);
printf("***");
position_cursor( 3, 6);
printf("SULPHUR WAVEFORM RECORDING");

/*
 Prompt the user for a list of the channels to plot.  We
 only allow up to 3 channels.
*/

position_cursor( 7, 0);

/* Get the number of channels to plot */

plot_count = 1;
```

```
/* Find out which channels to plot */

        plot_chan[ 0] = 0;

chan_count = 1;

/* Find out the operating input voltage range */

position_cursor(15, 0);
input_range = 1;

if( select == 1 )
{

/*  If we're getting new A/D data */

        /* Initialize the data acquisition boards */

        start_chan = 0;
        end_chan = 0;
        chan_count =1;

        status = 0;
        status += lp_set_error_control_word( 1);
        status += lp_initialize();
        status += lp_select_board( unit);
        status += lp_setup_adc( timing, start_chan, end_chan, gain);
        status += lp_set_clock_frequency( &freq);

        if( status != 0 )
        {

                printf("\nErrors  have  been  detected.   Do you\n");
                printf("want to continue?  (Y or N): ");
                ch = getch();
                while( (ch != 'Y') && (ch != 'y')
                        && (ch != 'N') && (ch != 'n'))
                {
                        BEEP;
                        ch = getch();
                }
                if( (ch == 'N') ¦¦ (ch == 'n') )
                {
                status = lp_terminate();
                return;
                }
        }

        status = lp_set_error_control_word( 0);
}
else
{

        eof_found = FALSE;

}


clear_text_window( 0, 0, 24, col_end, NORMAL_VIDEO);
```

```
        if( input_range == 1 )
        {

        /* Input range = 0 to +5 */

        /* Initialize the y locations of tic marks */

                tic_cnt = 6;
                for( i = 0, j = (ys + (CHAR_WIDTH / 2) - 2); i < 6; i++ )
                {
                        tic_location[ i] = j;
                        j = j + (CHAR_WIDTH * 4);
                }

        /* Label the y axis */

                position_cursor( 0, 0);
                printf("VOLTS");
                for( i = 1, j = 5; j >= 0; i += 4, j--)
                {
                        position_cursor( i, 0);
                        printf("%+id", j);
                }
        }

        /* range-specific markings are complete. now continue with the x axis */

        position_cursor( 22, 35);
        printf("TIME");

        /* Display the header information */

        for( i = 0; i < plot_count; i++)
        {
                position_cursor( 0, (10 + (10 * i)) );
                printf("File: %s",line);
                color_id = 3 - i;
                x1 = ((10 + 6 + (10 * i)) * CHAR_WIDTH) + 3;
                x2 = x1 + 20;
                y1 = CHAR_HEIGHT >> 1;
                y2 = y1;
    }

        /* Display the information at the bottom of the screen */

/*      position_cursor( 24, 0);
        printf("Press <ENTER> to stop or <ESC> to exit.");
*/
        /* Box in the plotting region */

        color_id = 3;

        /* left side */

        x1 = xs - 1;
        x2 = x1;
        y1 = ys;
        y2 = ye + 1;
        draw_line( x1, y1, x2, y2, color_id);
```

```
/* right side */

x1 = xe + 1;
x2 = x1;
y1 = ys;
y2 = ye + 1;
draw_line( x1, y1, x2, y2, color_id);

/* top side */

x1 = xs - 1;
x2 = xe + 1;
y1 = ys - 1;
y2 = y1;
draw_line( x1, y1, x2, y2, color_id);

/* bottom side */

x1 = xs - 1;
x2 = xe + 1;
y1 = ye + 1;
y2 = y1;
draw_line( x1, y1, x2, y2, color_id);

/*
 Draw the initial tracking bar which scans left to right across
 the screen.
*/

position = xs;
draw_line( (position + 1), ys, (position + 1), ye, 3);

/*
 This is the main acquisition and plotting loop.  Here we acquire the
 A/D data and plot it continuously until the user presses the <ESC> key.
*/

tic_counter = 0;
done = FALSE;
while( !done )
{
        /*
         Check for the valid key codes and perform desired
         actions if a valid key has been pressed.
        */

        while( check_standard_input() == TRUE )
        {  /*  Looking for valid input characters */
                if( (ch = getch()) == 0 )
                {  /*  Key is a special function, get 2nd key code */

                        ch = getch();
                        BEEP;
                }
                else if( ch == CR )
                {  /*  Carriage return means pause */

                        ch = getch();

                        if( ch == ESC )
                {  /*  Escape key means exit */
                        done = TRUE;
                        break;
                }
        }
        else if( ch == ESC )
        {  /*  Escape key means exit */
                done = TRUE;
                break;
        }
        else
        {  /*  Else the key pressed is not a valid option */
                BEEP;
        }
}

if( select == 1 )
{

/* get time of day */
time(&time_val);
i = fwrite((char*)&time_val,sizeof(long),
                1,stream);
if(i !=1)break;
rec_count += 4;
/*  If we're getting new A/D data */

        status = lp_adc_series( chan_count, adc_data);
        xadc_data[0] = 4095 - adc_data[0];
        i = fwrite((char*)xadc_data,sizeof(int),
                        1,stream);
        if (i != 1)break;
        rec_count += 2;
}
else
{

/*  Read in the time value */
        items_read = fread((char *)&time_val,
                sizeof(long), 1, stream);
        if( items_read != 1) eof_found = TRUE;

/*  Read in the A/D data */
        items_read = fread((char *)xadc_data, sizeof(int),
                1, stream);
        if( items_read != 1) eof_found = TRUE;


/*
 If end-of-file has been found then stop.
*/
        if( eof_found )
        {

        /*  EOF has been found, stop charting */
                position_cursor( 22, 0);
                printf("End of file found!");
                while( (ch = getch()) != ESC );
                return;
        }
```

```
                }

        /* Get the new points to plot */

                        value = ( long )xadc_data[ 0];
                        value = (value * ( long )(ye - ys + 1)) / 0x0FFFL;
                        if(select == 1)
                        {
/*                      new_value[ 0] = ye - (.69*(ye-ys) - ( int )value);
*/                      new_value[ 0] = ye - ( int )value;
                        }
                        else
                        {
                        new_value[0] = ye - (int)value;
                        }
                        if( new_value[ 0] < ys ) new_value[ 0] = ys;

        /* Check to see if we've come back to the beginning */

                if( position == xs )
                {
                                old_value[ 0] = new_value[ 0];
                }

        /*
         Update the tracking bar.  First draw new bar, then remove old bar.
         The order is important: this helps reduce unsightly flicker.
        */
                old_position = position;
                if( position++ == xe )
                {
                        position = xs;
                        old_position = xs - 1;
                }
                draw_line( (position + 1), ys, (position + 1), ye, 3);
                draw_line( (old_position + 1), ys, (old_position + 1), ye, 0);

        /*
         Plot the A/D values by drawing a line from last
         value read to the new value.
        */

                if( position == xs ) old_position = xs;
                for( i = 0; i < plot_count; i++)
                {
                        draw_line( old_position, old_value[ i],
                                position, new_value[ i], 3 - i);
                }

        /* Draw tic marks if necessary */

                if( ++tic_counter == 20 )
                {
                        tic_counter = 0;
                        for( i = 0; i < tic_cnt; i++ )
                        {
                                draw_pixel( old_position, tic_location[ i], 3);
                        }
                }
        /* We've plotted the point, now it's the old point */

                        old_value[ 0] = new_value[ 0];

        }

        if( select == 1 )
        {
                status = lp_terminate();
                rewind(stream);
                i = fwrite((char*)&rec_count,sizeof(int),1,stream);
        }

        return;

} /* strip_chart */
```

```
print_ansi_device_warning()
{
/*
 This function simply displays a warning message that the ANSI video device
 driver has been installed and will interfere with the intended screen
 appearance.
*/

        int     ch;             /* Input character from keyboard */

        BEEP;
        printf("\n");
        printf("The  device  driver  ANSI.SYS  has been ");
        printf("detected as being loaded as a memory\n");
        printf("resident  driver.   This  driver  ");
        printf("interferes  with   the   intended   screen\n");
        printf("attributes  of  this  program.   This  ");
        printf("program  will run correctly,  but the\n");
        printf("screen will not appear with  the  correct  ");
        printf("highlighting  and  reverse  video\n");
        printf("fields  that  were  intended.   For the ");
        printf("most pleasing screen appearance, the\n");
        printf("device driver ANSI.SYS should not be ");
        printf("loaded when using this program.\n");
        printf("\n");

        flushall();             /* Clear all current keyboard input */
        printf("Press any key to continue.\n");
        ch = getch();

        return;

}/* print_ansi_device_warning */
```

```
print_cga_missing_warning()
{
/*
 This function simply displays a warning message that the color graphics
 adapter was not found in the current system configuration.
*/
        printf("\n\nThis  program  cannot  be  run without a color ");
        printf("graphics adapter (CGA).  This\n");
        printf("program could not locate a CGA in your system.  ");
        printf("Program execution has ended.\n\n\n");

        return;

} /* print_cga_missing_warning */
```

```
print_column_size_warning()
{
/*
 This function simply displays a warning message that the column size of the
 display device is not 40 characters.

 Because this function uses the screen I/O routine clear_text_window,
 initialize_video must be called once before this function is called.
*/

        int     ch;             /* Input character from keyboard */

        BEEP;
        printf("\n");

        printf("This  program  was intended to be used\n");
        printf("on a  40 column  display.   The  video\n");
        printf("display  is  currently  configured for\n");
        printf("something other than 40 columns.  This\n");
        printf("program  will  still  work  correctly;\n");
        printf("however, the screen will not appear as\n");
        printf("it   was   intended.    For  the  most\n");
        printf("pleasing   screen   appearance,    the\n");
        printf("display  should  be  configured for 40\n");
        printf("columns  black/white  or  40   columns\n");
        printf("color.   For  information  on  how  to\n");
        printf("change  the   display   configuration,\n");
        printf("refer  to the MODE command in your DOS\n");
        printf("USER'S GUIDE.\n");
        printf("\n");

        flushall();             /* Clear all current keyboard input */
        printf("Press any key to continue.\n");
        ch = getch();

        clear_text_window( 0, 0, 24, col_end, NORMAL_VIDEO);

        return;

} /* print_column_size_warning */
```